

MATH GAME

Submitted in partial fulfillment of the requirements

Of the Second Year of Bachelor of Artificial Intelligence and Data Science

By

Mr. Shubham Uttam Bakade

Roll No. 50

Mr. Sahil Rushikant Mahadik

Roll No. 59

Mr. Sahil Jayram Raut

Roll No. 62

Mr. Shaikh Mohammad Sarfaraz Mohammad Ali

Roll No. 63

Project Guide

MRS. K.S. YADAV



DEPARTMENT OF ARTIFICIAL INTELLIGENCE & DATA SCIENCE

KONKAN GYANPEETH COLLEGE OF ENGINEERING

UNIVERSITY OF MUMBAI

(2024-2025)

CERTIFICATE

This is to certify the project entitled “**Math Game**” is a bonafide work of “**Shubham Bakade (50), Sahil Mahadik (59), Sahil Raut (62), Shaik Mohammad Sarfaraz (63)**” Submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the “**S.E.**” in “**Artificial Intelligence and Data Science**”.

(Mrs. K.S. Yadav)
Project Guide

(Prof. G. V. Dakhave)
Head of Department

(Dr.V.J.Pillewan)
Principal

KONKAN GYANPEETH COLLEGE OF ENGINEERING, KARJAT.

(Affiliated to University of Mumbai, Approved by A.I.C.T.E. ,New Delhi.

Konkan Gyanpeeth Shaikshanik Sankul, Vengaon Road, Dahivali, Karjat, Dist.-Raigad. 410201.(M.S.)

Department Of Artificial Intelligence and Data Science

Project Report Approval

This project report entitled “**Math Game**” by “**Shubham Bakade (50), Sahil Mahadik (59), Sahil Raut (62), Shaik Mohammad Sarfaraz (63)**” is approved for the S.E. of Artificial Intelligence and Data Science.

Examiners

1.....

2.....

Date.

Place.

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Mr. Shubham Bakade

Mr. Sahil Mahadik

Mr. Sahil Raut

Mr. Shaikh Mohammad Sarfaraz

(Name of student's and Roll No.)

(Signatures)

Date:

INDEX

TITLE	Pg. No.
ABSTRACT	i
LIST OF FIGURES	ii
1. INTRODUCTION	1-2
1.1 Introduction and Motivation	1
1.2 Objectives	1
1.3 Problem Statement	2
1.4 Scope of Project	2
2. REVIEW OF LITERATURE	3-5
3. REQUIREMENTS ANALYSIS	6-6
3.1 Software Requirements	6
3.2 Hardware Requirements	6
3.3 System Requirements	6
4. DESIGN AND PLANNING	7-9
4.1 Flow Chart	8
4.2 UI Design	9
5. IMPLEMENTATION	10-23
5.1 Code	10
5.2 Output	20
6. FUTURE ENHANCEMENT	24-24
7. CONCLUSION	25-25
8. REFERENCES	26-26

ABSTRACT

The provided HTML and JavaScript code creates a simple math quiz game where players answer randomly generated arithmetic questions within a 60-second time limit. The user interface consists of elements to display the current score, the question, answer choices, a countdown timer, and messages indicating whether the answers are correct or incorrect. When the player clicks the "Start Game" button, the game resets, and the timer begins, while new questions are generated for the player to solve.

As the game progresses, players can select their answers by clicking on one of the four provided boxes. The game logic checks the chosen answer against the correct one and updates the score accordingly. If the answer is correct, a success message appears briefly, and a new question is generated; if incorrect, a retry message is displayed. The game ends when the timer runs out, showing a "Game Over" message along with the player's final score, and allowing them to restart by clicking the button again.

This interactive quiz not only helps players practice basic arithmetic skills but also incorporates elements of game design, such as scoring and timed challenges, making learning engaging and fun. The use of JavaScript for game mechanics combined with HTML for structure provides a straightforward yet effective learning tool.

LIST OF FIGURES

Figure No.	Title	Pg. No.
Figure 4.1	Flow chart	8
Figure 4.2	UI Design	9
Figure 5.1	Initially Page	20
Figure 5.2	When Question is generated and right answer is selected	20
Figure 5.3	Wrong answer is selected	21
Figure 5.4	Right answer of Subtraction	21
Figure 5.5	Wrong answer of Multiplication	22
Figure 5.6	Right answer of Addition	22
Figure 5.7	Score get's displayed	23

CHAPTER 1

INTRODUCITON

1.1 Introduction and Motivation

JavaScript is a scripting or programming language that allows you to implement complex features on web pages. Every time a web page does more than just sit there and display static information for you to look at. Displaying timely content updates, interactive maps, animated 2D / 3D graphics, scrolling videos jukeboxes, etc. We have made a math game, this can improve other brain functions, such as attention, concentration, and focus. It helps the users in practicing Math. It is a Learn And Fun Game. It can be used by any Age-Groups. Math game give space to critical thinking and that helps children nurture their attention to detail while solving it. Math games can improve visual recognition as well as help in investing leisure time in something quite useful.

1.2 Objectives

1. To create an interactive web-based game for practicing basic arithmetic.
2. To improve the user's math problem-solving skills through immediate feedback.
3. To provide a user-friendly interface that caters to users of all ages.

1.3 Problem Statement

Learning math can sometimes be monotonous, especially for young students. There is a need for an engaging and interactive platform that can help users practice math in a fun way, providing immediate feedback and motivation.

1.4 Scope of Project

The scope of this project includes developing a basic maths quiz game covering arithmetic operations like addition, subtraction, multiplication, and division. It provides multiple-choice questions with a continuous flow of randomly generated questions. Future iterations of the project could include a timer, difficulty levels, and a scoring system.

CHAPTER 2

REVIEW OF LITERATURE

Sr. No.	Title	Author	Year of Publication	Description
1.	A Case Study on 21st Century Skills Development Through a Computer Based Maths Game	Nour El Mawas, Michael Bradford, Josephine Andrews, Pramod Pathak, Cristina Hava Muntean	2019	<p>1. New tech developments require re-skilling and lifelong learning, especially in 21st century skills.</p> <p>2. Education should focus on preparing learners with problem solving, self-directed learning, and digital literacy.</p> <p>3. Mathematics is often challenging for students, especially in developing logical thinking and problem-solving skills.</p> <p>4. "Count With Me!" is a novel educational video game designed to teach math concepts and promote 21st century skills.</p>

2.	The potential of an interactive game-based software to motivate high-achieving maths students at primary school level	Stephen Howard	2021	<p>1. Systematic review of 13 studies on game-based approaches to reduce Maths Anxiety (MA) in children and young adults.</p> <p>2. Games used were mainly general learning (quiz/puzzle-based).</p> <p>3. 8/13 studies showed reduced MA after game-based interventions, but only 6/13 showed significant reduction compared to traditional learning.</p> <p>4. Few games were designed explicitly for MA reduction; only 2 studies included anxiety-aware features (e.g., competition element, real-time MA indicator).</p> <p>5. Review highlights the need for targeted studies to design games addressing MA, including gender-aware and collaborative game features.</p>
----	---	----------------	------	--

CHAPTER 3

SYSTEM REQUIREMENTS

3.1 Software Requirements

1. HTML5
2. CSS3
3. JavaScript
4. Web browser (e.g., Chrome, Firefox)

3.2 Hardware Requirements

1. Any device that supports a modern web browser (e.g., PC, Laptop, Tablet)

3.3 System Requirements

1. Windows, macOS, or Linux operating system
2. Stable internet connection

CHAPTER 4

DESIGN AND PLANNING

4.1 Flow Chart

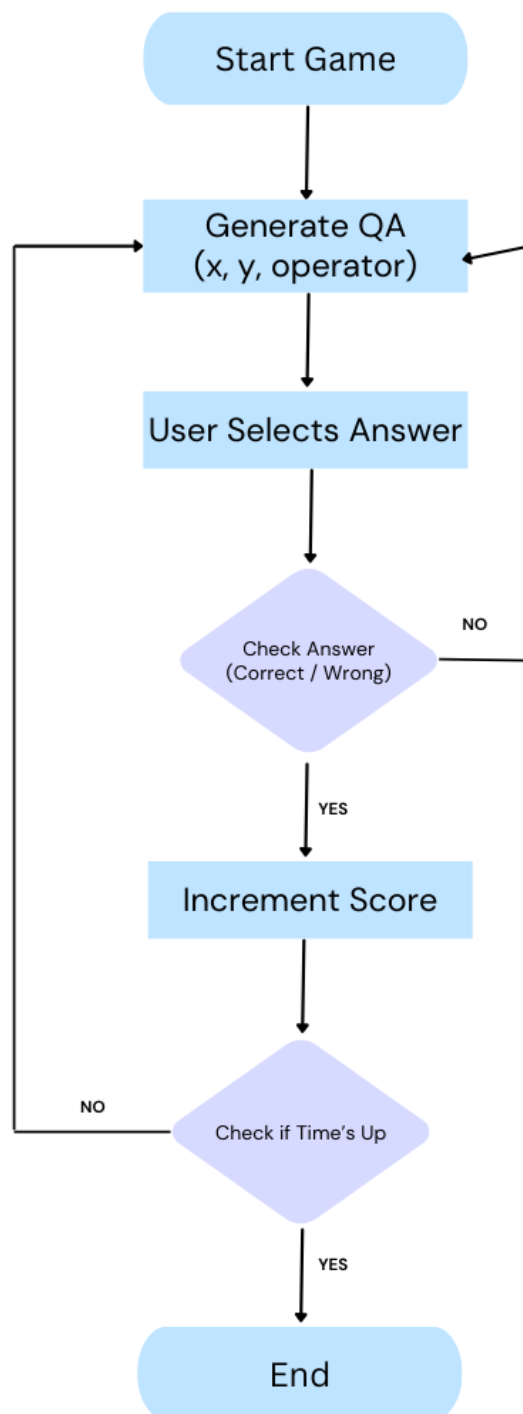


Fig 4.1 Flow chart

4.2 UI Design

The UI design for this code includes the following elements:

1. **Score Display:** Shows the current score with a heading "Score" and the value.
2. **Feedback Messages:** Displays "Correct" for correct answers and "Try again" for wrong answers, which appear and disappear based on user input.
3. **Question Display:** Shows the math question to be answered.
4. **Instruction:** A small note telling the user to click on one of the answers below.
5. **Answer Boxes:** Four clickable boxes (box1, box2, box3, box4) where one holds the correct answer, and the others hold wrong answers.
6. **Start/Reset Button:** Starts the game or resets it if already playing, toggling between "Start Game" and "Reset Game."
7. **Timer:** Shows the remaining time in seconds, starting from 60.
8. **Game Over Screen:** Displays a game-over message and the final score when time runs out.

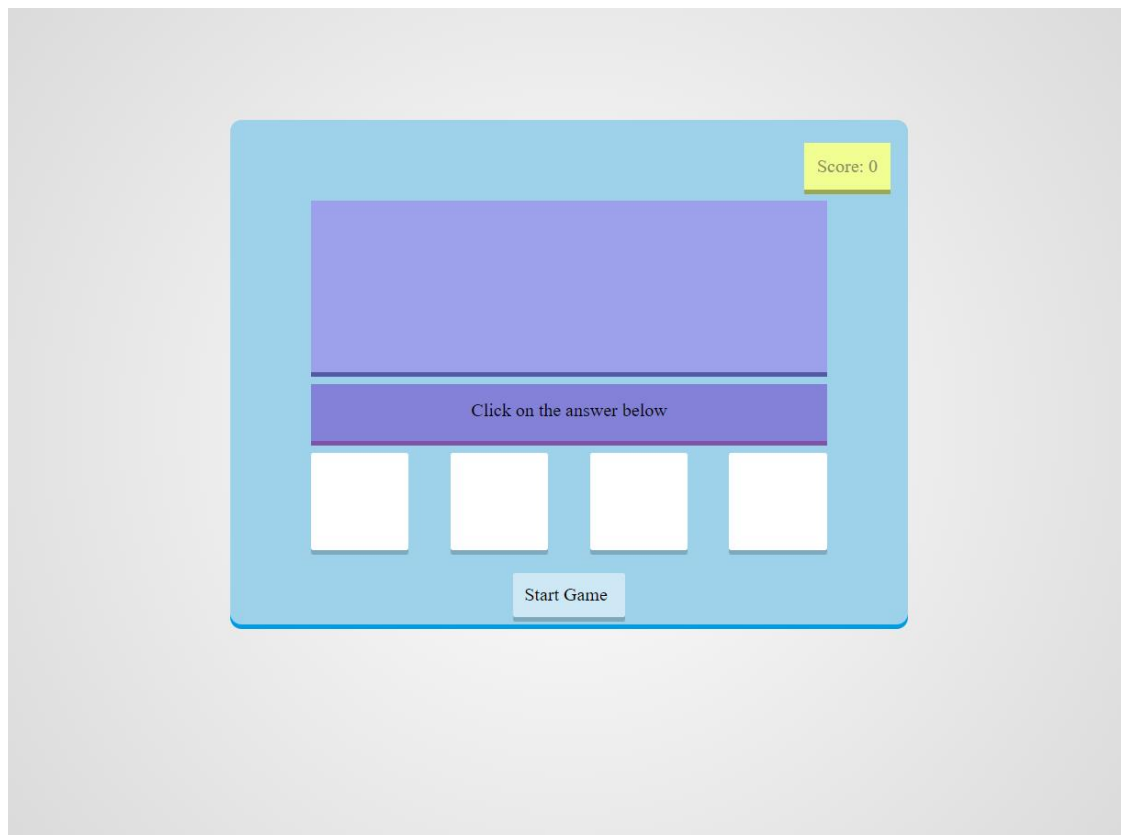


Fig 4.2 UI Design

CHAPTER 5

IMPLEMENTATION

5.1 CODE

index.html

```
<!DOCTYPE html>
<html lang="en">

<head>
  <title>Math Game</title>
  <style>
    html {
      height: 100%;
      background: radial-gradient(circle, #fff, #ccc)
    }

    * {
      margin: 0;
      padding: 0;
    }
```

```
#container {  
    background: #9dd2ea;  
    width: 550px;  
    height: 400px;  
    margin: 100px auto;  
    border-radius: 10px;  
    padding: 20px;  
    box-shadow: 0px 4px 0px 0px #009de4;  
    -moz-box-shadow: 0px 4px 0px 0px #009de4;  
    -webkit-box-shadow: 0px 4px 0px 0px #009de4;  
    position: relative;  
}
```

```
#score {  
    background-color: #f1ff92;  
    color: #888e5f;  
    padding: 11px;  
    position: absolute;  
    left: 500px;  
    box-shadow: 0px 4px 0px 0px #9da853;  
}
```

```
#correct {  
    background-color: green;  
    color: white;  
    padding: 11px;  
    position: absolute;  
    left: 260px;  
    display: none;  
}
```

```
#wrong {  
    background-color: red;  
    color: white;
```

```

padding: 11px;
position: absolute;
left: 260px;
display: none;
}

#question {
width: 450px;
height: 150px;
margin: 50px auto 10px auto;
background-color: #9da0ea;
box-shadow: 0px 4px #535aa8;
font-size: 100px;
text-align: center;
font-family: cursive, sans-serif;
}

#instruction {
width: 450px;
height: 50px;
margin: 10px auto;
background: #8481D9;
text-align: center;
line-height: 45px;
box-shadow: 0px 4px #8153a8;
}

#choices {
width: 450px;
height: 150px;
margin: 5px auto;
}

.box {
background-color: white;

```

```
width: 85px;
height: 85px;
float: left;
margin-right: 36.5px;
border-radius: 2px;
cursor: pointer;
box-shadow: 0px 4px rgba(0, 0, 0, 0.2);
text-align: center;
line-height: 80px;
position: relative;
transition: all 0.2s;
-webkit-transition: all 0.2s;
-o-transition: all 0.2s;
-ms-transition: all 0.2s;
-moz-transition: all 0.2s;
}
```

```
.box:hover,
#startReset:hover {
background-color: #9c89f6;
color: white;
box-shadow: 0px 4px #6b54de;
-webkit-box-shadow: 0px 4px #6b54de;
-moz-box-shadow: 0px 4px #6b54de;
}
```

```
.box:active,
#startReset:active {
box-shadow: 0px 0px;
-moz-box-shadow: 0px 0px;
-webkit-box-shadow: 0px 0px;
top: 4px;
}
```

```
#box4 {
```

```

        margin-right: 0;
    }

    #startReset {
        background-color: rgba(255, 255, 255, 0.5);
        width: 78px;
        border: none;
        padding: 10px;
        border-radius: 2px;
        cursor: pointer;
        box-shadow: 0px 4px rgba(0, 0, 0, 0.2);
        position: relative;
        transition: all 0.2s;
        -webkit-transition: all 0.2s;
        -o-transition: all 0.2s;
        -ms-transition: all 0.2s;
        -moz-transition: all 0.2s;
        margin: 0 auto;
        margin-top: -50px;
    }

    #timerremaining {
        width: 152px;
        padding: 10px;
        position: absolute;
        top: 395px;
        left: 400px;
        background-color: rgba(181, 235, 36, 0.78);
        border-radius: 3px;
        box-shadow: 0px 4px rgba(0, 0, 0, 0.2);
        -webkit-box-shadow: 0px 4px rgba(0, 0, 0, 0.2);
        -moz-box-shadow: 0px 4px rgba(0, 0, 0, 0.2);
        display: none;
    }

```

```

#gameover {
  height: 200px;
  width: 500px;
  text-align: center;
  color: white;
  position: absolute;
  top: 100px;
  left: 40px;
  font-size: 2.5em;
  background: linear-gradient(#f3ca6b, #fe706c);
  z-index: 2;
  display: none;
}
</style>
</head>

<body>
  <div id="container">
    <div id="score">Score: <span id="scorevalue">0</span></div>
    <div id="correct">Correct</div>
    <div id="wrong">Try again</div>
    <div id="question"></div>
    <div id="instruction"> Click on the answer below</div>
    <div id="choices">
      <div class="box" id="box1"></div>
      <div class="box" id="box2"></div>
      <div class="box" id="box3"></div>
      <div class="box" id="box4"></div>
    </div>
    <div id="startReset">Start Game</div>
    <div id="timerremaining">Time remaining: <span id="timerremainingvalue">60</span>
sec</div>
    <div id="gameover"></div>
  </div>
  <script src="script1.js"></script>

```

</body>

</html>

script1.js

```
let playing = false;
let timerremaining;
let score;
let x, y, z, z1, choiceRandom, operator;
let startTime; // To track the start time

document.getElementById('startReset').onclick = function () {
  if (playing == true) {
    location.reload();
  } else {
    playing = true;
    hide("gameover");
    show("timerremaining");
    document.getElementById('startReset').innerHTML = "Reset Game";
    score = 0;
    document.getElementById('scorevalue').innerHTML = score;
    startTime = Date.now(); // Record the starting time
    countdown();
    generateQA();
  }
}

function countdown() {
  let countdownInterval = setInterval(function () {
    let elapsed = Math.floor((Date.now() - startTime) / 1000); // Time elapsed in seconds
    timerremaining = 60 - elapsed;
    document.getElementById('timerremainingvalue').innerHTML = timerremaining;

    if (timerremaining <= 0) {
```

```

        stopcountdown(countdownInterval);
        show("gameover");
        document.getElementById('startReset').innerHTML = "Start Game";
        document.getElementById("gameover").innerHTML = "<p>GAME
OVER</p><br><p>YOUR SCORE IS " + score + "</p>";
        playing = false;
    }
}, 100); // Check the elapsed time frequently
}

function stopcountdown(interval) {
    clearInterval(interval);
}

function generateQA() {
    x = Math.round(1 + Math.random() * 9);
    y = Math.round(1 + Math.random() * 9);

    // Randomly select an operator
    let operators = ['+', '-', 'x'];
    operator = operators[Math.floor(Math.random() * operators.length)];

    if (operator == '+') {
        z = x + y;
    } else if (operator == '-') {
        z = x - y;
    } else {
        z = x * y;
    }

    document.getElementById("question").innerHTML = x + ' ' + operator + ' ' + y;

    // Place the correct answer in a random box
    choiceRandom = Math.round(1 + Math.random() * 3);
    document.getElementById('box' + choiceRandom).innerHTML = z;

```



```

let wrongAnswers = [z];

for (let i = 1; i < 5; i++) {
  if (i !== choiceRandom) {
    do {
      x = Math.round(1 + Math.random() * 9);
      y = Math.round(1 + Math.random() * 9);

      if (operator === '+') {
        z1 = x + y;
      } else if (operator === '-') {
        z1 = x - y;
      } else {
        z1 = x * y;
      }
    } while (wrongAnswers.indexOf(z1) > -1);

    wrongAnswers.push(z1);
    document.getElementById('box' + i).innerHTML = z1;
  }
}

for (let i = 1; i < 5; i++) {
  document.getElementById("box" + i).onclick = function () {
    if (playing === true) {
      if (this.innerHTML === z) {
        show("correct");
        hide("wrong");
        setTimeout(function () {
          hide("correct");
          hide("wrong");
        }, 1000);
        score++;
        document.getElementById('scorevalue').innerHTML = score;
      }
    }
  }
}

```

```
        generateQA();
    } else {
        hide("correct");
        show("wrong");
        setTimeout(function () {
            hide("wrong");
        }, 1000);
    }
}
}
}

function show(id) {
    document.getElementById(id).style.display = "block";
}

function hide(id) {
    document.getElementById(id).style.display = "none";
}
```

5.2 OUTPUT

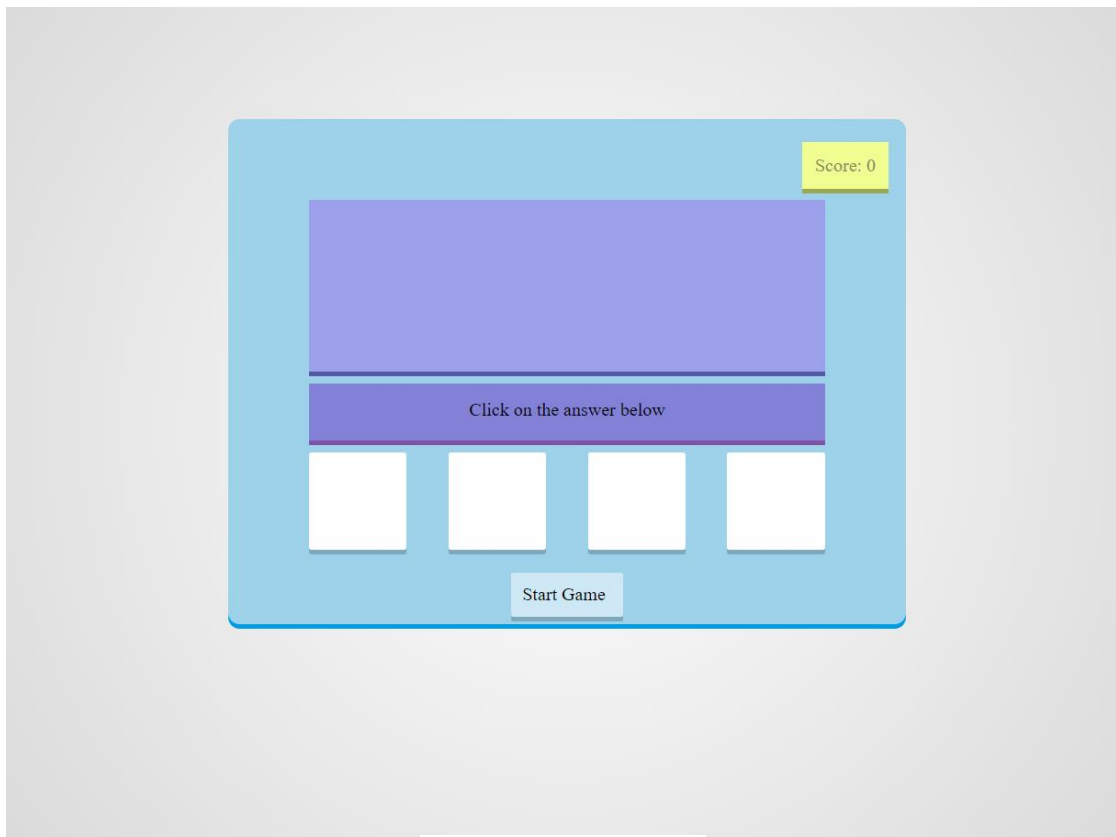


Fig. 5.1 Initially page

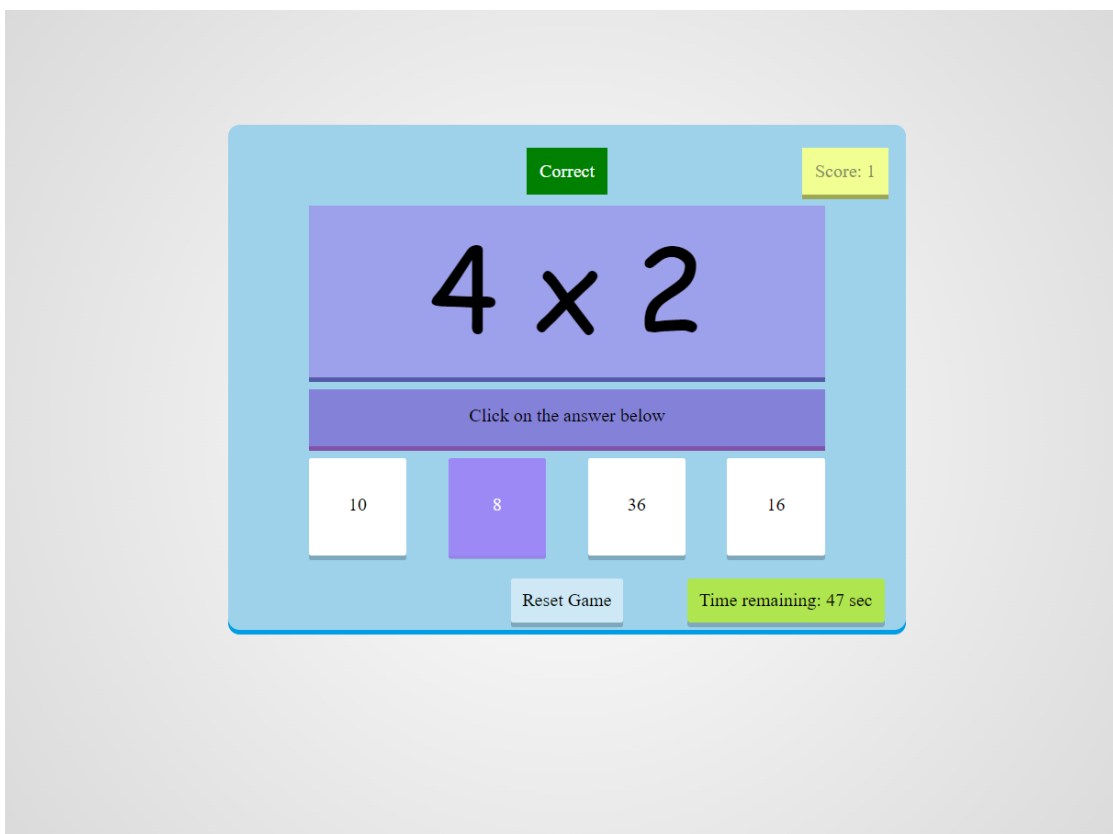


Fig. 5.2 When Question is generated and correct answer is selected

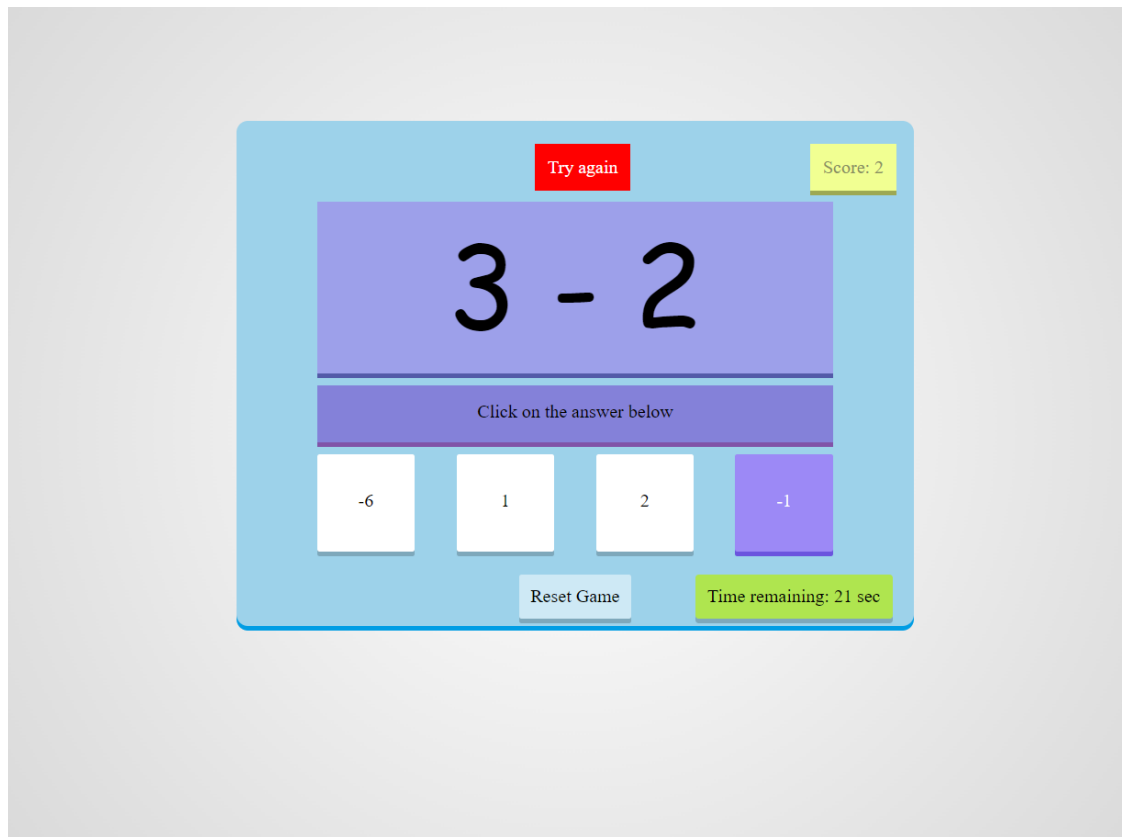


Fig. 5.3 Wrong answer is selected

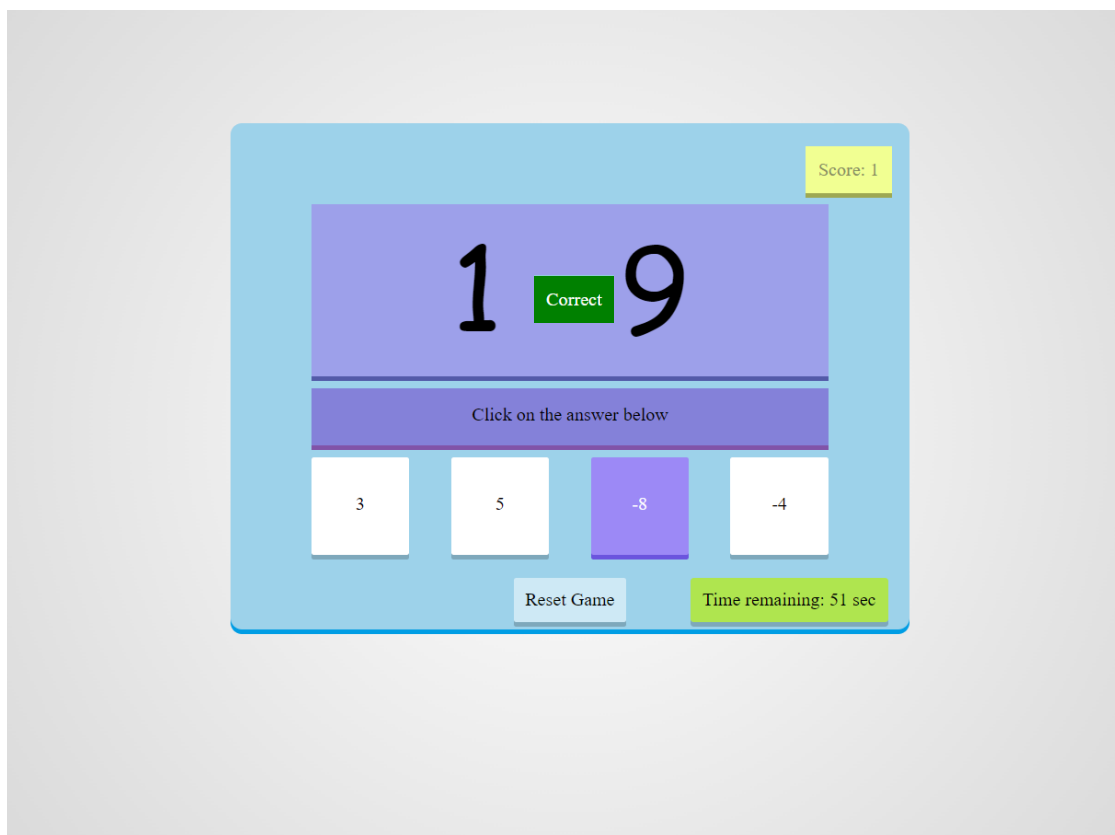


Fig. 5.4 Right answer of Subtraction

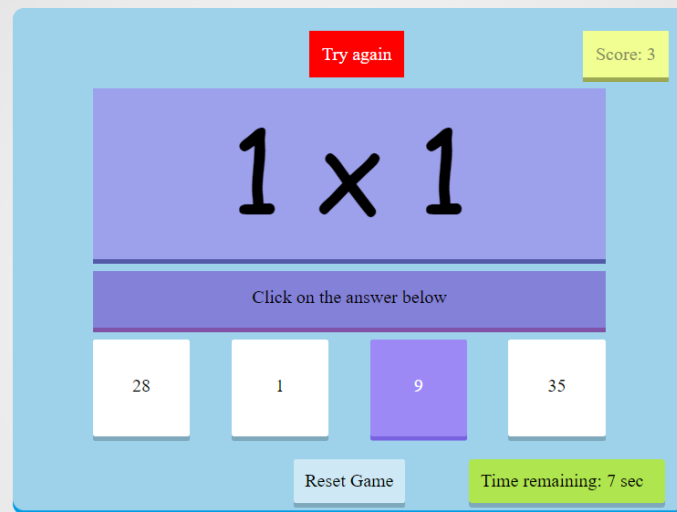


Fig. 5.5 Wrong answer of Multiplication

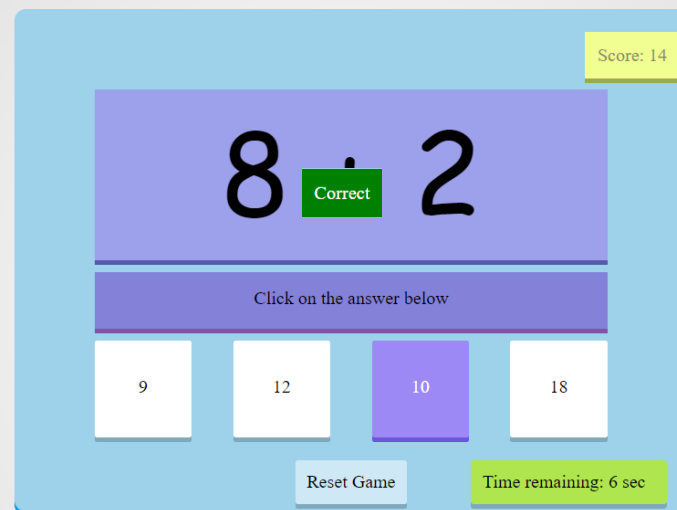


Fig. 5.6 Right answer of Addition

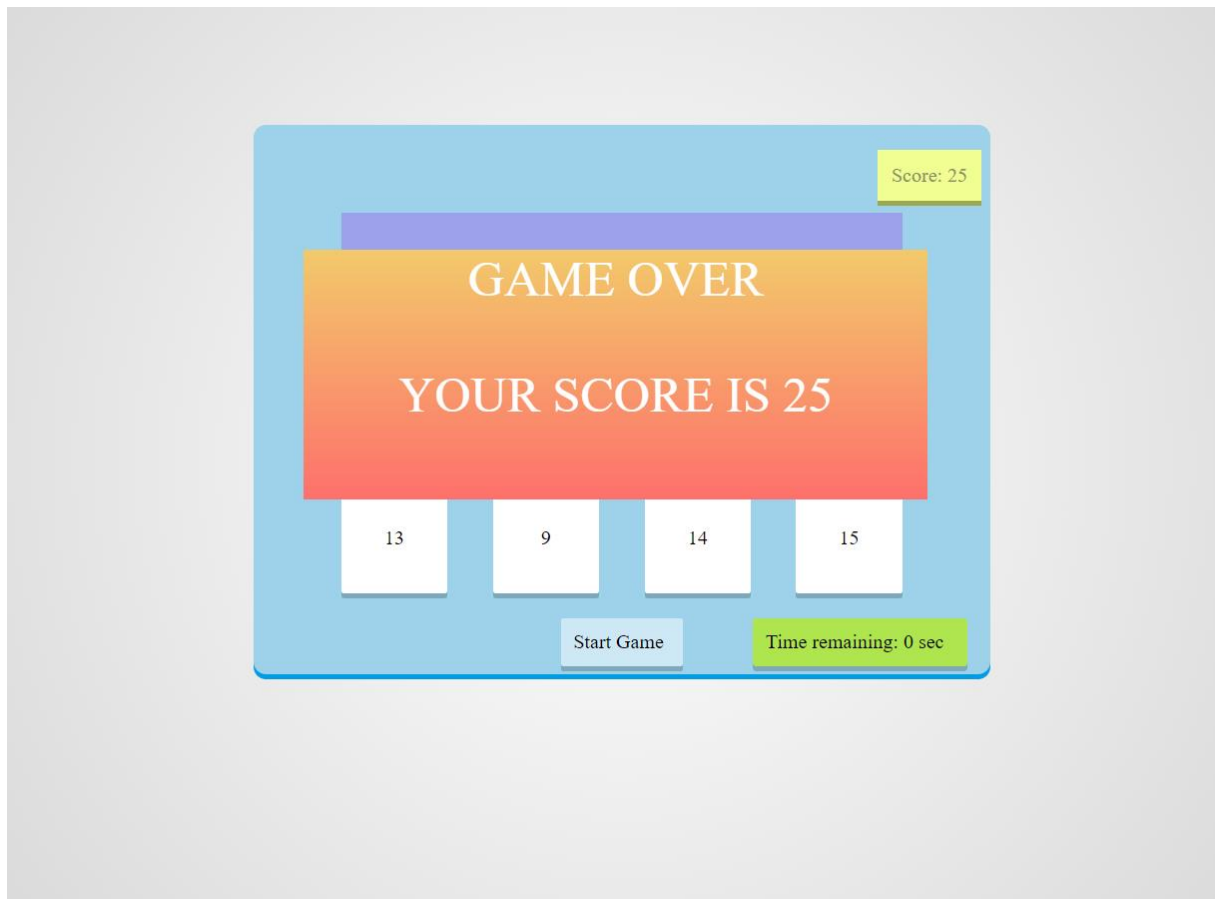


Fig. 5.7 Score get's displayed

CHAPTER 6

FUTURE ENHANCEMENT

Future enhancements to the project could include:

1. Implementing different difficulty levels (easy, medium, hard).
2. Expanding the range of math operations covered.
3. Adding more sums as per to jr. collages.

CHAPTER 7

CONCLUSION

This code creates a simple math quiz game in HTML, CSS, and JavaScript. The game generates random arithmetic questions (addition, subtraction, or multiplication) and presents four possible answers, with one being correct. The user must click on the correct answer, and their score increases for each correct response. The game also includes a 60-second countdown timer, and when the time runs out, the game ends and displays the user's final score. The player can restart or reset the game using the "Start Game" or "Reset Game" button. The game dynamically handles correct and incorrect answers by showing corresponding feedback ("Correct" or "Try again") for a brief moment.

CHAPTER 8

REFERENCE

Referred book:

1. "Eloquent JavaScript: A Modern Introduction to Programming" by Marijn Haverbeke
2. "JavaScript: The Definitive Guide" by David Flanagan
3. "HTML & CSS: Design and Build Websites" by Jon Duckett