NORTH SOUTH UNIVERSITY

Department of Electrical & Computer Engineering

CSE299: Junior Design

Project Report

(Group-1)

Project Title: Web-Based Educational Game for Children

Submitted By	Submitted To

Member 1:
    Name: Md. Rakibul Hasan
    ID: 2111127642
Member 2:
    Name: Yeamin Hassan
    ID:2131637642
	Faculty: Omar Ibne Shahid
	
	North South University, Dhaka, Bangladesh

Introduction:

In today's digital age, educational games play a crucial role in enhancing children's learning experiences. As part of our CSE 299 course, we developed a web-based educational game designed to make learning fun and interactive for young students. This project combines education with entertainment, helping children develop essential skills in an engaging way. The game focuses on fundamental subjects such as mathematics, language, and problem-solving, targeting primary school students. Our goal is to increase learning motivation and retention by using gamification components.

Objective:

1. To create an interactive and user-friendly educational game for children.
2. To enhance learning through gamification techniques.
3. To develop a responsive web application with a colorful background.
4. To implement a secure database for storing user ID and Data.
5. To encourage self-paced learning in a fun and engaging manner.

Implementation:

Technical Details:

Frontend**:** HTML, CSS, JavaScript
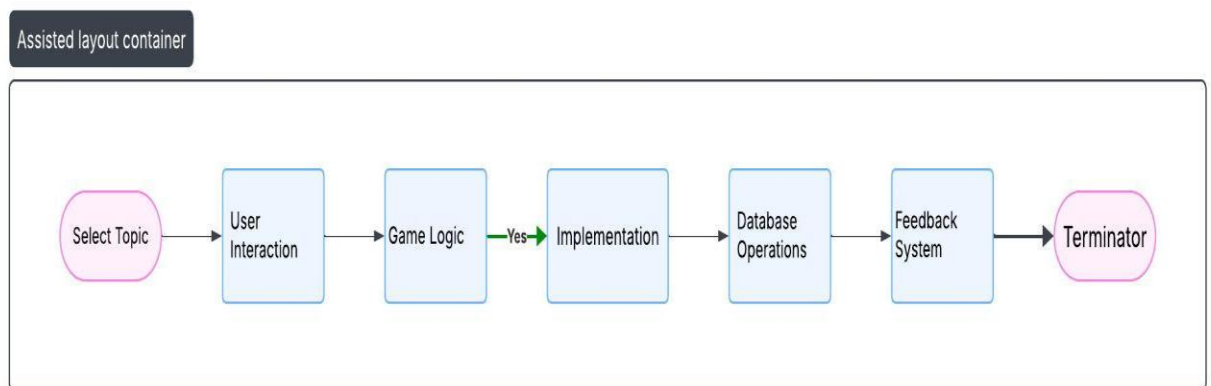
Backend**:** PHP

Database**:** MySQL

Process Diagram:

1. **User Interaction:** The child interacts with the game interface

2. **Game Logic:** We find the game logic of how this game should be interactive for the children.

3**. Implementation:** Frontend: HTML, CSS, JavaScript.

>>> Backend: PHP

>>> Database: MySQL

4**. Database Operations:** The backend stores User ID and User Data.

5. **Feedback System:** The game provides feedback and wishes.

Assisted layout container

Select Topic → User Interaction → Game Logic → Yes → Implementation → Database Operations → Feedback System → Terminator

Screenshots:

Codes:

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

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Game Home</title>
  <script src="https://cdn.tailwindcss.com"></script>
</head>

<body class="text-white w-full h-full bg-no-repeat bg-cover"
  style="background-image: url('./images/-Slidesdocs-childrens\ day\ children\ playing\ games_cba6ed0c08.jpg');
  ">
  <div class="h-screen flex flex-col items-center justify-center text-center p-6">
    <h1 class="text-5xl font-extrabold bg-gradient-to-r from-blue-500 to-purple-500 text-transparent bg-clip-text">
      Welcome!</h1>
    <p class="text-lg mt-4 text-black">Learn with fun!</p>

    <div class="mt-6 grid grid-cols-1 space-y-4">
      <a href="./login.html"
          class="px-6 py-3 bg-blue-500 text-white font-bold rounded-lg shadow-lg hover:bg-blue-400
          transition">Login</a>
      <a href="./signup.html"
          class="px-6 py-3 bg-blue-500 text-white font-bold rounded-lg shadow-lg hover:bg-blue-400
          transition">Sign
          Up</a>
    </div>
  </div>
</body>

</html>
```

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

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Game Dashboard</title>
  <script src="https://cdn.tailwindcss.com"></script>
</head>

<body class="text-white backdrop-blur-2xl w-full h-full bg-no-repeat bg-cover"
  style="background-image: url('./images/vecteezy_nature-landscape-vector-illustration-with-a-cute-and_13549621-1.jpg');">
  <div class="flex h-screen">
    <main class="flex-1 p-6">
      <div class="grid lg:grid-cols-4 md:grid-cols-2 gap-6 ">
        <a href="./science_game.html">
          <div class="bg-gradient-to-r from-purple-500 to-indigo-500 p-5 rounded-xl shadow-lg">
            <h2 class="text-lg font-semibold">Science</h2>
            <img src="./images/atom-laboratory-science-icon.png" class="w-full mt-2" alt="science">
          </div>
        </a>
        <a href="./math.html">
          <div class="bg-gradient-to-r from-red-500 to-pink-500 p-5 rounded-xl shadow-lg">
            <h2 class="text-lg font-semibold">Math</h2>
            <img src="./images/calculate-math-icon.png" class="w-full mt-2" alt="calculate-math-icon">
          </div>
        </a>
        <a href="./language.html">
          <div class="bg-gradient-to-r from-yellow-500 to-orange-500 p-5 rounded-xl shadow-lg">
            <h2 class="text-lg font-semibold">Language</h2>
            <img src="./images/language-icon.png" class="w-full mt-2" alt="language-icon">
          </div>
        </a>
        <a href="./puzzle.html">
          <div class="bg-gradient-to-r from-green-500 to-teal-500 p-5 rounded-xl shadow-lg">
            <h2 class="text-lg font-semibold">Puzzle & Logic</h2>
            <img src="./images/puzzle-icon.png" class="w-full mt-2" alt="puzzle-icon">
          </div>
        </a>
      </div>
    </main>
  </div>
</body>

</html>
```

chemistry.html

```html
<html lang="en">
<head>
</head>

<body class="flex flex-col items-center justify-center h-screen bg-gradient-to-r from-orange-100 via-green-400 to-green-700">
    <h1 id="create-text" class="text-3xl font-bold mb-2">Create:</h1>
    <p id="target-name" class="text-xl font-bold text-blue-800 mb-4"></p>

    <div id="experiment-container" class="center-container"></div>

    <div class="beaker bg-gray-300 w-32 h-40 relative mt-10" id="result-beaker">
        <div class="liquid" id="result-liquid" style="height: 0%;"></div>
    </div>

    <p id="reaction" class="mt-4 text-lg font-bold text-green-600 hidden"></p>
    <p id="wrong-message" class="mt-4 text-lg font-bold text-red-600 hidden">Wrong reaction!</p>

    <button id="nextStep" class="mt-4 px-4 py-2 bg-gray-500 text-white rounded disabled:opacity-50" disabled>Next</button>

    <!-- Game Over Section -->
    <div id="game-over" class="mt-4 text-xl font-bold text-purple-800 hidden flex flex-col items-center justify-center">
        <p>Correct Reactions: <span id="correct-count">0</span></p>
        <a href="./dashboard.html"><button id="homeButton" class="mt-2 px-4 py-2 bg-gray-500 text-white rounded">Home</button></a>
    </div>

    <script>
        let experiments = [
            { reactants: ['H₂', 'O₂'], product: 'H₂O', equation: '2H₂ + O₂ → 2H₂O', color: 'bg-blue-400' },
            { reactants: ['Na', 'Cl₂'], product: 'NaCl', equation: '2Na + Cl₂ → 2NaCl', color: 'bg-yellow-400' },
            { reactants: ['C', 'O₂'], product: 'CO₂', equation: 'C + O₂ → CO₂', color: 'bg-gray-500' },
            { reactants: ['Fe', 'O₂'], product: 'Fe₂O₃', equation: '4Fe + 3O₂ → 2Fe₂O₃', color: 'bg-red-500' },
            { reactants: ['Ca', 'CO₂'], product: 'CaCO₃', equation: 'Ca + CO₂ → CaCO₃', color: 'bg-gray-400' }
        ];

        let correctCount = 0;
        let currentExperiment = 0;
        const container = document.getElementById('experiment-container');
        const targetName = document.getElementById('target-name');
        const createText = document.getElementById('create-text');
        const reactionText = document.getElementById('reaction');
        const wrongMessage = document.getElementById('wrong-message');
        const nextButton = document.getElementById('nextStep');
        const resultBeaker = document.getElementById('result-beaker');
        const resultLiquid = document.getElementById('result-liquid');
        const gameOverDiv = document.getElementById('game-over');
        const correctCountSpan = document.getElementById('correct-count');
        const homeButton = document.getElementById('homeButton');
```

---

math.html

```html
<html lang="en">
<head>
    <style>
    </style>
</head>

<body class="flex flex-col items-center justify-center h-screen bg-gradient-to-r from-orange-100 via-green-400 to-green-700">
    <div id="game-container" class=" p-8 rounded-lg shadow-2xl text-center w-96">
        <p id="question" class="text-2xl font-bold mb-4 text-gray-800"></p>
        <div id="options" class="grid grid-cols-2 gap-4 mb-4"></div>
        <p id="result" class="mt-4 text-xl font-bold"></p>
        <button id="next" class="mt-4 bg-green-500 text-white px-4 py-2 rounded-lg hover:bg-green-600 hidden">Next</button>
    </div>

    <script>
        let currentProblem = 0;
        let problems = [];
        let score = 0;

        function generateProblem() {
            const num1 = Math.floor(Math.random() * 10) + 1;
            const num2 = Math.floor(Math.random() * 10) + 1;
            const operators = ['+', '-', '*', '/'];
            const operator = operators[Math.floor(Math.random() * operators.length)];

            let answer;
            switch (operator) {
                case '+':
                    answer = num1 + num2;
                    break;
                case '-':
                    answer = num1 - num2;
                    break;
                case '*':
                    answer = num1 * num2;
                    break;
                case '/':
                    answer = (num1 / num2).toFixed(2);
                    break;
            }

            const wrongAnswers = [];
            while (wrongAnswers.length < 3) {
                const wrongAnswer = Math.floor(Math.random() * 20) + 1;
                if (wrongAnswer !== answer && !wrongAnswers.includes(wrongAnswer)) {
                    wrongAnswers.push(wrongAnswer);
                }
            }
```

dashboard.html chemistry.html index.html language.html science_game.html biology.html puzzle.html × physics.html test.html

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Puzzle Game</title>
  <script src="https://cdn.tailwindcss.com"></script>
  <style>
    @keyframes pop {
      0% { transform: scale(1); }
      50% { transform: scale(1.1); }
      100% { transform: scale(1); }
    }
    .tile {
      transition: transform 0.2s ease, background-color 0.2s ease;
    }
    .tile:hover {
      transform: scale(1.05);
    }
    .win-animation {
      animation: pop 0.5s ease;
    }
  </style>
</head>
<body class="flex items-center justify-center h-screen w-full h-full bg-no-repeat bg-cover bg-gradient-to-r from-orange-100 via-green-400 to-green-700">
  <div class="text-center">
    <h1 class="text-4xl font-bold mb-6 text-white">Number Puzzle</h1>
    <div id="puzzle-grid" class="grid grid-cols-3 gap-3 w-72 h-72 mx-auto mb-6"></div>
    <div class="flex flex-col gap-6">
      <button id="shuffle-btn" class="bg-yellow-400 text-white px-6 py--3 rounded-full text-lg font-bold hover:bg-yellow-500 transition-all">
        Shuffle
      </button>
      <a href="./dashboard.html">
        <button class="bg-gray-400 text-white px-6 py-3 rounded-full text-lg font-bold hover:bg-gray-500 transition-all">
          Home
        </button>
      </a>
    </div>
    <p id="message" class="mt-6 text-2xl font-bold text-white"></p>
  </div>

  <script>
    const gridSize = 3;
    let tiles = [];
    let emptyTile = { row: gridSize - 1, col: gridSize - 1 };

    function initializePuzzle() {
```

---

dashboard.html signup.php × index.html language.html science_game.html biology.html physics.html test.html

```php
<?php
$host = 'localhost';
$dbname = 'myapp';
$username = 'root';
$password = '';
try {
    $conn = new PDO("mysql:host=$host;dbname=$dbname", $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch (PDOException $e) {
    die("Database connection failed: " . $e->getMessage());
}

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    // Get form data
    $username = $_POST['username'];
    $email = $_POST['email'];
    $password = $_POST['password'];
    $confirmPassword = $_POST['confirm-password'];

    if ($password !== $confirmPassword) {
        die("Passwords do not match.");
    }

    $hashedPassword = password_hash($password, PASSWORD_DEFAULT);

    try {
        $sql = "INSERT INTO users (username, email, password) VALUES (:username, :email, :password)";
        $stmt = $conn->prepare($sql);
        $stmt->bindParam(':username', $username);
        $stmt->bindParam(':email', $email);
        $stmt->bindParam(':password', $hashedPassword);
        $stmt->execute();


        $url = "./login.html";
        header('Location: '.$url);



    } catch (PDOException $e) {
        die("Error registering user: " . $e->getMessage());
    }
}
?>
```

```php
<?php
$host = 'localhost';
$dbname = 'myapp';
$username = 'root';
$password = '';

try {
    $conn = new PDO("mysql:host=$host;dbname=$dbname", $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch (PDOException $e) {
    die("Database connection failed: " . $e->getMessage());
}

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $email = $_POST['email'];
    $password = $_POST['password'];

    try {
        $sql = "SELECT * FROM users WHERE email = :email";
        $stmt = $conn->prepare($sql);
        $stmt->bindParam(':email', $email);
        $stmt->execute();

        if ($stmt->rowCount() > 0) {
            $user = $stmt->fetch(PDO::FETCH_ASSOC);

            if (password_verify($password, $user['password'])) {
                header("Location: dashboard.html");
                exit();
            } else {
                header("Location: login.html?error=Incorrect password");
                exit();
            }
        } else {
            header("Location: login.html?error=User not found");
            exit();
        }
    } catch (PDOException $e) {
        die("Error fetching user: " . $e->getMessage());
    }
}
?>
```

Database:

Challenges:

During development, we faced several challenges:

1. **Balancing Education and Fun:** Ensuring the game was both educational and enjoyable.

2. **Responsive Design:** Making the game work smoothly on different screen sizes.

3. **Database Security:** Protecting user data.

Limitations:

1. Limited number of games due to time constraints.

2. Basic graphics due to limited resources.

3. No multiplayer or social features.

Future Work:

1. More games will be added.

2. The games will be more graphical.

3. Multiplayer feature will be added.

Discussion:

Our web-based educational game successfully demonstrates how technology can make learning more engaging for children. The integration of gamification elements helps maintain interest, while the responsive design ensures accessibility. However, further testing with real users (children and educators) could provide valuable feedback for improvements.

Conclusion:

This project was a valuable learning experience in full-stack web development and educational technology. We successfully built a functional game that combines education and entertainment with the potential for future enhancements. By continuing to refine the application, we can create an even more effective tool for children's learning.

GitHub: https://github.com/rakibulhasanme09/CSE299_Project.git