```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>TechQuest: Aventura Digital PRO</title>
  <style>
    body {
      font-family: 'Segoe UI', sans-serif;
      background-color: #121212;
      color: #f5f5f5;
      text-align: center;
    }

    #game-container {
      max-width: 600px;
      margin: 20px auto;
      padding: 20px;
      background-color: #1e1e2f;
      border-radius: 10px;
      box-shadow: 0 0 10px #0ff;
    }

    canvas {
      background-color: #333;
      margin-top: 10px;
      border: 2px solid #0ff;
    }

    button {
      background-color: #0ff;
      border: none;
      color: #000;
      padding: 8px 16px;
      margin: 8px;
      border-radius: 5px;
      font-weight: bold;
      cursor: pointer;
    }

    #info, #question-area, #gameCanvas, #end-screen {
      display: none;
    }

    input[type="text"] {
      padding: 8px;
      width: 60%;
      border-radius: 4px;
      margin-top: 10px;
```

```
    }

    #feedback {
      margin-top: 10px;
      font-style: italic;
    }
  </style>
</head>
<body>
  <div id="game-container">
    <div id="start-screen">
      <h1>TechQuest: Aventura Digital PRO</h1>
      <p>¡Bienvenido a la aventura tecnológica!<br>
      Mueve al jugador con las flechas del teclado y recoge monedas.<br>
      Responde preguntas para avanzar niveles. Usa tus puntos para comprar pistas.<br>
      ¡Buena suerte!</p>
      <button onclick="startGame()">Comenzar</button>
    </div>

    <canvas id="gameCanvas" width="300" height="300"></canvas>

    <div id="info">
      <p>Nivel: <span id="level">1</span> | Puntos: <span id="score">0</span></p>
      <button onclick="buyHint()">Comprar pista (3 pts)</button>
      <p id="hint"></p>
    </div>

    <div id="question-area">
      <p><strong>Pregunta:</strong> <span id="question-text"></span></p>
      <input type="text" id="answer-input" placeholder="Escribe tu respuesta...">
      <button onclick="checkAnswer()">Responder</button>
      <p id="feedback"></p>
    </div>

    <div id="end-screen">
      <h2>¡Juego terminado!</h2>
      <p>Puntaje final: <span id="final-score"></span></p>
      <p>Respuestas correctas: <span id="correct-count"></span></p>
      <p>Respuestas incorrectas: <span id="incorrect-count"></span></p>
      <p>Pistas usadas: <span id="hints-used"></span></p>
      <button onclick="location.reload()">Jugar de nuevo</button>
    </div>
  </div>

  <script>
    const canvas = document.getElementById('gameCanvas');
    const ctx = canvas.getContext('2d');
    const gridSize = 6;
```

```
    const cellSize = canvas.width / gridSize;
    let player = { x: 0, y: 0 };
    let coins = [];
    let score = 0;
    let level = 0;
    let correctAnswers = 0;
    let incorrectAnswers = 0;
    let hintIndex = 0;
    let hintsUsed = 0;

    const levels = [
      {
        question: "¿Qué es una computadora?",
        answer: "una máquina",
        hints: ["Empieza con 'una m...'", "Se usa para tareas digitales", "Tiene teclado y
pantalla"]
      },
      {
        question: "¿Qué es un mouse?",
        answer: "dispositivo",
        hints: ["Sirve para mover el cursor", "Tiene botones", "Empieza con 'd'"]
      },
      {
        question: "¿Qué es Internet?",
        answer: "una red",
        hints: ["Conecta computadoras", "Se usa para navegar", "Empieza con 'una r...'"]
      }
    ];

    function startGame() {
      document.getElementById('start-screen').style.display = 'none';
      canvas.style.display = 'block';
      document.getElementById('info').style.display = 'block';
      document.getElementById('question-area').style.display = 'block';
      nextLevel();
    }

    function nextLevel() {
      if (level >= levels.length) {
        endGame();
        return;
      }

      document.getElementById('hint').textContent = '';
      hintIndex = 0;

      // Generar monedas aleatorias
      coins = [];
```

```
  while (coins.length < 5) {
    let coin = {
      x: Math.floor(Math.random() * gridSize),
      y: Math.floor(Math.random() * gridSize)
    };
    if (!(coin.x === player.x && coin.y === player.y)) {
      coins.push(coin);
    }
  }

  player = { x: 0, y: 0 };
  drawGame();

  document.getElementById('level').textContent = level + 1;
  document.getElementById('question-text').textContent = levels[level].question;
}

function drawGame() {
  ctx.clearRect(0, 0, canvas.width, canvas.height);
  ctx.strokeStyle = "#666";
  for (let i = 0; i <= gridSize; i++) {
    ctx.beginPath();
    ctx.moveTo(i * cellSize, 0);
    ctx.lineTo(i * cellSize, canvas.height);
    ctx.stroke();
    ctx.beginPath();
    ctx.moveTo(0, i * cellSize);
    ctx.lineTo(canvas.width, i * cellSize);
    ctx.stroke();
  }

  for (const coin of coins) {
    ctx.fillStyle = "yellow";
    ctx.beginPath();
    ctx.arc(
      coin.x * cellSize + cellSize / 2,
      coin.y * cellSize + cellSize / 2,
      10,
      0,
      2 * Math.PI
    );
    ctx.fill();
  }

  ctx.fillStyle = "cyan";
  ctx.fillRect(
    player.x * cellSize + 5,
    player.y * cellSize + 5,
```

```
    cellSize - 10,
    cellSize - 10
  );
}

function updateScore() {
  document.getElementById("score").textContent = score;
}

function buyHint() {
  if (score >= 3 && hintIndex < levels[level].hints.length) {
    score -= 3;
    updateScore();
    document.getElementById("hint").textContent += levels[level].hints[hintIndex++] + " ";
    hintsUsed++;
  } else if (hintIndex >= levels[level].hints.length) {
    document.getElementById("hint").textContent += "(No hay más pistas)";
  } else {
    alert("No tienes suficientes puntos.");
  }
}

function checkForCoins() {
  for (let i = 0; i < coins.length; i++) {
    if (player.x === coins[i].x && player.y === coins[i].y) {
      coins.splice(i, 1);
      score++;
      updateScore();
      break;
    }
  }
}

function movePlayer(dx, dy) {
  const newX = player.x + dx;
  const newY = player.y + dy;
  if (newX >= 0 && newX < gridSize && newY >= 0 && newY < gridSize) {
    player.x = newX;
    player.y = newY;
    checkForCoins();
    drawGame();
  }
}

function checkAnswer() {
  const input = document.getElementById("answer-input").value.trim().toLowerCase();
  const correct = levels[level].answer.toLowerCase();
  const feedback = document.getElementById("feedback");
```

```
    if (input.includes(correct)) {
      feedback.textContent = "¡Correcto! 🎉";
      feedback.style.color = "#0f0";
      correctAnswers++;
      level++;
      setTimeout(() => {
        feedback.textContent = "";
        document.getElementById("answer-input").value = "";
        nextLevel();
      }, 1000);
    } else {
      feedback.textContent = "Incorrecto, intenta de nuevo.";
      feedback.style.color = "#f33";
      incorrectAnswers++;
    }
  }

  function endGame() {
    document.getElementById("gameCanvas").style.display = "none";
    document.getElementById("info").style.display = "none";
    document.getElementById("question-area").style.display = "none";
    document.getElementById("end-screen").style.display = "block";
    document.getElementById("final-score").textContent = score;
    document.getElementById("correct-count").textContent = correctAnswers;
    document.getElementById("incorrect-count").textContent = incorrectAnswers;
    document.getElementById("hints-used").textContent = hintsUsed;
  }

  document.addEventListener("keydown", (e) => {
    switch (e.key) {
      case "ArrowUp": movePlayer(0, -1); break;
      case "ArrowDown": movePlayer(0, 1); break;
      case "ArrowLeft": movePlayer(-1, 0); break;
      case "ArrowRight": movePlayer(1, 0); break;
    }
  });

  updateScore();
 </script>
</body>
</html>
```