

```
<!DOCTYPE html>
<html lang="pt-BR">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Curso de Desenvolvimento de Jogos 2D - GameCreators</title>
  <style>
    @import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;600;700&display=swa
p');
```

```
:root {
  --roxo: #6C5CE7;
  --ciano: #00CEC9;
  --amarelo: #FDCB6E;
  --branco: #F9F9F9;
  --cinza-escuro: #2D3436;
  --azul-escuro: #1E1E2F;
}
```

```
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: 'Poppins', sans-serif;
}
```

```
body {
  background-color: var(--branco);
  color: var(--cinza-escuro);
  line-height: 1.6;
}
```

```
.container {
  max-width: 1200px;
  margin: 0 auto;
  padding: 20px;
  position: relative;
  min-height: 100vh;
}
```

```
header {
  background: linear-gradient(135deg, var(--roxo) 0%, var(--ciano) 100%);
  color: var(--branco);
  padding: 25px;
  border-radius: 15px;
  margin-bottom: 30px;
  text-align: center;
```

```
    box-shadow: 0 8px 25px rgba(108, 92, 231, 0.3);  
}
```

```
header h1 {  
    font-size: 2.8rem;  
    margin-bottom: 10px;  
    text-shadow: 2px 2px 4px rgba(0,0,0,0.3);  
}
```

```
header p {  
    font-size: 1.3rem;  
    opacity: 0.95;  
}
```

```
.menu-aulas {  
    background-color: var(--azul-escuro);  
    position: sticky;  
    top: 20px;  
    z-index: 100;  
    padding: 20px;  
    border-radius: 15px;  
    margin-bottom: 30px;  
    display: flex;  
    justify-content: center;  
    flex-wrap: wrap;  
    gap: 12px;  
    box-shadow: 0 6px 20px rgba(30, 30, 47, 0.4);  
}
```

```
.menu-aulas button {  
    background-color: var(--roxo);  
    color: var(--branco);  
    border: none;  
    padding: 12px 18px;  
    border-radius: 8px;  
    cursor: pointer;  
    font-weight: 600;  
    font-size: 0.9rem;  
    transition: all 0.3s ease;  
    min-width: 80px;  
}
```

```
.menu-aulas button:hover {  
    background-color: var(--ciano);  
    transform: translateY(-3px);  
}
```

```
.menu-aulas button.active {
```

```

    background-color: var(--ciano);
    box-shadow: 0 0 15px rgba(0, 206, 201, 0.6);
}

.aula {
    display: none;
    background-color: white;
    border-radius: 15px;
    padding: 35px;
    margin-bottom: 30px;
    box-shadow: 0 8px 25px rgba(0, 0, 0, 0.12);
}

.aula.active {
    display: block;
}

.aula-header {
    text-align: center;
    margin-bottom: 30px;
}

.aula-header h2 {
    color: var(--roxo);
    font-size: 2.2rem;
    margin-bottom: 20px;
    padding-bottom: 15px;
    border-bottom: 4px solid var(--amarelo);
    display: inline-block;
}

.btn-estela {
    background: linear-gradient(135deg, var(--roxo) 0%, var(--ciano) 100%);
    color: white;
    border: none;
    padding: 15px 25px;
    border-radius: 10px;
    cursor: pointer;
    font-weight: 600;
    font-size: 1.1rem;
    margin: 10px;
    transition: all 0.3s ease;
    box-shadow: 0 4px 15px rgba(108, 92, 231, 0.3);
}

.btn-estela:hover {
    transform: translateY(-2px);
    box-shadow: 0 6px 20px rgba(108, 92, 231, 0.4);
}

```

```
}
```

```
.texto-estela {  
  background: linear-gradient(135deg, rgba(108, 92, 231, 0.1) 0%, rgba(0, 206, 201,  
0.1) 100%);  
  border: 2px solid var(--ciano);  
  border-radius: 15px;  
  padding: 25px;  
  margin: 20px 0;  
  display: none;  
  font-size: 1.1rem;  
  line-height: 1.8;  
}
```

```
.texto-estela.ativo {  
  display: block;  
  animation: slideIn 0.5s ease-out;  
}
```

```
.texto-estela h4 {  
  color: var(--roxo);  
  margin-bottom: 15px;  
  font-size: 1.3rem;  
}
```

```
.audio-controls {  
  text-align: center;  
  margin: 20px 0;  
  padding: 15px;  
  background-color: rgba(108, 92, 231, 0.1);  
  border-radius: 10px;  
}
```

```
.audio-controls button {  
  background-color: var(--roxo);  
  color: white;  
  border: none;  
  padding: 10px 15px;  
  border-radius: 5px;  
  cursor: pointer;  
  margin: 5px;  
  font-weight: 500;  
  transition: background-color 0.3s ease;  
}
```

```
.audio-controls button:hover {  
  background-color: var(--ciano);  
}
```

```
.audio-controls button.active {
  background-color: var(--ciano);
}

.secao {
  margin-bottom: 35px;
}

.secao h3 {
  color: var(--roxo);
  margin-bottom: 20px;
  font-size: 1.6rem;
}

.secao p {
  margin-bottom: 18px;
  font-size: 1.15rem;
  line-height: 1.7;
}

.destaque {
  background: linear-gradient(135deg, rgba(108, 92, 231, 0.1) 0%, rgba(0, 206, 201,
0.1) 100%);
  border-left: 5px solid var(--roxo);
  padding: 20px;
  margin: 25px 0;
  border-radius: 8px;
  font-weight: 500;
}

.exemplo {
  background-color: rgba(0, 206, 201, 0.12);
  border: 2px solid var(--ciano);
  padding: 25px;
  border-radius: 12px;
  margin: 25px 0;
}

.exemplo h4 {
  color: var(--ciano);
  margin-bottom: 15px;
  font-size: 1.3rem;
}

.quiz {
  background: linear-gradient(135deg, rgba(253, 203, 110, 0.2) 0%, rgba(253, 203,
110, 0.1) 100%);
```

```
border: 2px solid var(--amarelo);
padding: 25px;
border-radius: 12px;
margin: 30px 0;
}

.quiz h4 {
  color: var(--cinza-escuro);
  margin-bottom: 20px;
  font-size: 1.3rem;
}

.opcoes {
  display: grid;
  gap: 12px;
  margin: 20px 0;
}

.opcao {
  background-color: white;
  border: 2px solid #ddd;
  padding: 15px;
  border-radius: 8px;
  cursor: pointer;
  transition: all 0.3s ease;
  font-weight: 500;
}

.opcao:hover {
  background-color: rgba(108, 92, 231, 0.1);
  border-color: var(--roxo);
  transform: translateX(5px);
}

.opcao.correta {
  background-color: rgba(46, 213, 115, 0.2);
  border-color: #2ed573;
}

.opcao.incorreta {
  background-color: rgba(255, 71, 87, 0.2);
  border-color: #ff4757;
}

.navegacao {
  display: flex;
  justify-content: space-between;
  margin-top: 40px;
```

```
padding-top: 25px;
border-top: 2px solid #eee;
}
```

```
.navegacao button {
  background: linear-gradient(135deg, var(--roxo) 0%, var(--ciano) 100%);
  color: white;
  border: none;
  padding: 15px 30px;
  border-radius: 8px;
  cursor: pointer;
  font-weight: 600;
  font-size: 1rem;
  transition: all 0.3s ease;
}
```

```
.navegacao button:hover {
  transform: translateY(-3px);
  box-shadow: 0 6px 20px rgba(108, 92, 231, 0.4);
}
```

```
.navegacao button:disabled {
  background: #ccc;
  cursor: not-allowed;
  transform: none;
  box-shadow: none;
}
```

```
.status-audio {
  margin: 10px 0;
  padding: 10px;
  background: rgba(0, 206, 201, 0.1);
  border-radius: 5px;
  font-size: 0.9rem;
  color: var(--cinza-escuro);
  display: none;
}
```

```
.status-audio.show {
  display: block;
}
```

```
@keyframes slideIn {
  from {
    opacity: 0;
    transform: translateY(20px);
  }
  to {
```

```

        opacity: 1;
        transform: translateY(0);
    }
}

.hidden { display: none; }

/* Responsividade - CORRIGIDO para manter botões lado a lado */
@media (max-width: 768px) {
    .container { padding: 15px; }
    header h1 { font-size: 2.2rem; }
    .aula { padding: 25px; }
    .aula-header h2 { font-size: 1.8rem; }

    /* REMOVIDO: flex-direction: column; para manter botões lado a lado */
    .menu-aulas {
        gap: 8px; /* Reduzido para caber melhor */
    }

    .menu-aulas button {
        padding: 10px 14px; /* Botões menores em mobile */
        font-size: 0.8rem;
        min-width: 70px;
    }

    .navegacao {
        flex-direction: column;
        gap: 15px;
    }
}

@media (max-width: 480px) {
    .menu-aulas button {
        padding: 8px 12px;
        font-size: 0.75rem;
        min-width: 60px;
    }
}
</style>
</head>
<body>
    <div class="container">
        <!-- Header Principal -->
        <header>
            <h1>🎮 Desenvolvimento de Jogos 2D</h1>
            <p>Aprenda a criar seu próprio jogo do zero com a GameCreators!</p>
        </header>

```


<!-- Menu de Navegação das Aulas -->

<nav class="menu-aulas">

<button class="menu-btn active" onclick="irParaAula(1)">Aula 1</button>

<button class="menu-btn" onclick="irParaAula(2)">Aula 2</button>

<button class="menu-btn" onclick="irParaAula(3)">Aula 3</button>

<button class="menu-btn" onclick="irParaAula(4)">Aula 4</button>

<button class="menu-btn" onclick="irParaAula(5)">Aula 5</button>

<button class="menu-btn" onclick="irParaAula(6)">Aula 6</button>

<button class="menu-btn" onclick="irParaAula(7)">Aula 7</button>

<button class="menu-btn" onclick="irParaAula(8)">Aula 8</button>

</nav>

<!-- Aula 1: Introdução aos Princípios de Design de Jogos -->

<section class="aula active" id="aula1">

<div class="aula-header">

<h2>🎯 Aula 1: Princípios de Design de Jogos</h2>

<button class="btn-estela" onclick="ouvirEstela(1)">



Ouvir explicação da Estela

</button>

<div class="status-audio" id="status-audio-1"></div>

<div class="texto-estela" id="texto-estela-1">

<h4>🤖 Estela está explicando:</h4>

<p>Olá! Eu sou a Estela, sua assistente virtual especializada em desenvolvimento de jogos! Seja muito bem-vindo ao nosso curso de Desenvolvimento de Jogos 2D!</p>

<p>Nesta primeira aula, vamos mergulhar nos fundamentos do design de jogos. Design de jogos é muito mais do que apenas criar algo divertido - é uma arte que combina psicologia, tecnologia e criatividade para criar experiências memoráveis.</p>

<p>Vamos começar entendendo os 4 pilares fundamentais:
Mecânica (as regras do seu jogo), Estética (como seu jogo se apresenta visualmente e sonoramente), Tecnologia (as ferramentas que você vai usar), e História (que dá significado às ações do jogador).</p>

<p>O conceito mais importante que quero que você entenda hoje é o "fluxo" - aquele estado onde o jogador fica completamente imerso no jogo, perdendo a noção do tempo. Isso acontece quando há um equilíbrio perfeito entre desafio e habilidade.</p>

<div class="audio-controls">

<button onclick="pararAudio()">🔴 Parar</button>

<button onclick="pausarAudio()">⏸ Pausar</button>

<button onclick="continuarAudio()">▶ Continuar</button>

<label>Velocidade: </label>

```
<button onclick="mudarVelocidade(0.5)">0.5x</button>
<button onclick="mudarVelocidade(1)" class="active">1x</button>
<button onclick="mudarVelocidade(1.5)">1.5x</button>
<button onclick="mudarVelocidade(2)">2x</button>
</div>
</div>
</div>
```

```
<div class="secao">
  <h3>🎮 O que é Design de Jogos?</h3>
  <p><strong>Imagine o design de jogos como uma receita mágica!</strong> Você
mistura diversão, desafio e recompensa para criar uma experiência viciante.</p>
```

```
  <div class="destaque">
    <p>💡 <strong>Regra de Ouro:</strong> Um bom jogo equilibra desafio e
habilidade. Muito fácil = tédio. Muito difícil = frustração. O ponto doce é o
<strong>FLUXO</strong>!</p>
  </div>
</div>
```

```
<div class="quiz">
  <h4>📝 Mini Quiz: Teste seu conhecimento!</h4>
  <p>Qual elemento é essencial para manter o jogador no estado de "fluxo"?</p>
```

```
  <div class="opcoes">
    <div class="opcao" onclick="responder(this, false)">Gráficos perfeitos</div>
    <div class="opcao" onclick="responder(this, true)">Equilíbrio entre desafio e
habilidade</div>
    <div class="opcao" onclick="responder(this, false)">História complexa</div>
    <div class="opcao" onclick="responder(this, false)">Muitos personagens</div>
  </div>
</div>
```

```
<div class="navegacao">
  <button disabled>⬅ Voltar</button>
  <button onclick="proximaAula()">Próxima aula ➡</button>
</div>
</section>
```

```
<!-- Aula 2: Tipos de jogos e gêneros -->
<section class="aula" id="aula2">
  <div class="aula-header">
    <h2>🎮 Aula 2: Gêneros de Jogos 2D</h2>
    <button class="btn-estela" onclick="ouvirEstela(2)">
      🔊 Ouvir explicação da Estela
    </button>

    <div class="status-audio" id="status-audio-2"></div>
```

```
<div class="texto-estela" id="texto-estela-2">
  <h4>🤖 Estela está explicando:</h4>
  <p>Oba! Chegamos à aula dos gêneros de jogos! Esta é uma das minhas
partes favoritas porque vamos explorar um universo de possibilidades criativas.</p>
```

```
  <p>Cada gênero é como um sabor diferente de sorvete - alguns são clássicos
que nunca saem de moda, outros são inovações que criam tendências! Jogos de plataforma
focam em movimentação precisa, puzzles estimulam sua mente, RPGs permitem evoluir e
crescer, e shoot'em ups testam seus reflexos.</p>
```

```
  <p>A dica de ouro é escolher um gênero que você realmente ama jogar,
porque essa paixão vai te motivar durante todo o desenvolvimento. Lembre-se: você pode
misturar gêneros para criar algo único!</p>
```

```

  <div class="audio-controls">
    <button onclick="pararAudio()">🔴 Parar</button>
    <button onclick="pausarAudio()">⏸ Pausar</button>
    <button onclick="continuarAudio()">▶ Continuar</button>
    <br>
    <label>Velocidade: </label>
    <button onclick="mudarVelocidade(0.5)">0.5x</button>
    <button onclick="mudarVelocidade(1)" class="active">1x</button>
    <button onclick="mudarVelocidade(1.5)">1.5x</button>
    <button onclick="mudarVelocidade(2)">2x</button>
  </div>
</div>
</div>
```

```

<div class="secao">
  <h3>🌟 Universo de Possibilidades</h3>
  <p><strong>Cada gênero é como um sabor diferente de sorvete!</strong> Alguns
são clássicos, outros são inovadores. Vamos descobrir qual combina com você.</p>
</div>
```

```

<div class="quiz">
  <h4>📝 Quiz: Qual é seu gênero?</h4>
  <p>Qual gênero é perfeito para quem ama resolver problemas lógicos?</p>
```

```

  <div class="opcoes">
    <div class="opcao" onclick="responder(this, false)">Plataforma</div>
    <div class="opcao" onclick="responder(this, true)">Puzzle</div>
    <div class="opcao" onclick="responder(this, false)">Shooter</div>
    <div class="opcao" onclick="responder(this, false)">RPG</div>
  </div>
</div>
```

```
<div class="navegacao">
```

```
<button onclick="aulaAnterior()">⏪ Voltar</button>
<button onclick="proximaAula()">Próxima aula ⏩</button>
</div>
</section>
```

```
<!-- Demais aulas seguem o mesmo padrão... -->
<!-- Aula 3 -->
```

```
<section class="aula" id="aula3">
  <div class="aula-header">
    <h2>⚡ Aula 3: Framework MDA</h2>
    <button class="btn-estela" onclick="ouvirEstela(3)">
      🔊 Ouvir explicação da Estela
    </button>
```

```
<div class="status-audio" id="status-audio-3"></div>
```

```
<div class="texto-estela" id="texto-estela-3">
  <h4>🗣️ Estela está explicando:</h4>
  <p>Agora vamos mergulhar no framework MDA - Mecânica, Dinâmica e
Estética! Imagine que você está construindo uma máquina de diversão incrível.</p>
```

<p>As engrenagens são suas mecânicas - as regras que você cria. O movimento da máquina é a dinâmica - como os jogadores se comportam com suas regras. E a alegria que a máquina causa é a estética - as emoções que despertamos!</p>

<p>É um processo fascinante: você cria mecânicas simples, mas quando os jogadores interagem com elas, surgem comportamentos complexos e inesperados que geram emoções poderosas!</p>

```
<div class="audio-controls">
  <button onclick="pararAudio()">⏏ Parar</button>
  <button onclick="pausarAudio()">⏸ Pausar</button>
  <button onclick="continuarAudio()">▶ Continuar</button>
  <br>
  <label>Velocidade: </label>
  <button onclick="mudarVelocidade(0.5)">0.5x</button>
  <button onclick="mudarVelocidade(1)" class="active">1x</button>
  <button onclick="mudarVelocidade(1.5)">1.5x</button>
  <button onclick="mudarVelocidade(2)">2x</button>
</div>
</div>
</div>
```

```
<div class="secao">
  <h3>🔗 O Triângulo Mágico MDA</h3>
  <p><strong>Imagine que você está construindo uma máquina de
diversão!</strong> As engrenagens são as MECÂNICAS, o movimento é a DINÂMICA, e a
alegria que causa é a ESTÉTICA.</p>
```

```
</div>
```

```
<div class="navegacao">
```

```
  <button onclick="aulaAnterior()"><img alt="seta para esquerda" data-bbox="475 138 495 152"/> Voltar</button>
```

```
  <button onclick="proximaAula()">Próxima aula <img alt="seta para direita" data-bbox="588 155 608 169"/></button>
```

```
</div>
```

```
</section>
```

```
<!-- Continuar com as demais aulas... -->
```

```
<!-- Aula 4-8 omitidas para economizar espaço, mas seguem o mesmo padrão -->
```

```
</div>
```

```
<script>
```

```
  // ===== VARIÁVEIS GLOBAIS =====
```

```
  let aulaAtual = 1;
```

```
  let audioAtual = null;
```

```
  let velocidadeAtual = 1;
```

```
  // ===== CONFIGURAÇÃO DE ÁUDIOS PARA GITHUB PAGES =====
```

```
  const audiosEstela = {
```

```
    1: './audios/aula1.mp3',
```

```
    2: './audios/aula2.mp3',
```

```
    3: './audios/aula3.mp3',
```

```
    4: './audios/aula4.mp3',
```

```
    5: './audios/aula5.mp3',
```

```
    6: './audios/aula6.mp3',
```

```
    7: './audios/aula7.mp3',
```

```
    8: './audios/aula8.mp3'
```

```
  };
```

```
  // ===== FUNÇÃO PRINCIPAL - OUVIR ESTELA =====
```

```
  function ouvirEstela(numeroAula) {
```

```
    console.log('🎵 Iniciando explicação da aula:', numeroAula);
```

```
    // 1. SEMPRE mostrar o texto primeiro
```

```
    const textoDiv = document.getElementById('texto-estela-' + numeroAula);
```

```
    const statusDiv = document.getElementById('status-audio-' + numeroAula);
```

```
    if (textoDiv) {
```

```
      textoDiv.classList.add('ativo');
```

```
      showStatus(statusDiv, '📖 Texto carregado! Preparando áudio...', 'info');
```

```
      console.log('✅ Texto exibido para aula', numeroAula);
```

```
    } else {
```

```
      console.log('❌ Elemento texto não encontrado:', 'texto-estela-' + numeroAula);
```

```
      alert('❌ Erro: Elemento de texto não encontrado.');
```

```
      return;
```

```
    }
```

```

// 2. Tentar carregar e reproduzir áudio
const audioUrl = audiosEstela[numeroAula];
if (!audioUrl) {
  showStatus(statusDiv, '⚠️ Áudio não configurado para esta aula. Texto disponível
para leitura.', 'warning');
  console.log('⚠️ URL do áudio não encontrada para aula', numeroAula);
  return;
}

// 3. Parar áudio anterior se existir
if (audioAtual) {
  audioAtual.pause();
  audioAtual = null;
  console.log('🔴 Áudio anterior parado');
}

// 4. Criar e configurar novo áudio
try {
  showStatus(statusDiv, '🔄 Carregando áudio...', 'info');

  audioAtual = new Audio(audioUrl);
  audioAtual.playbackRate = velocidadeAtual;

  // Event listeners para feedback ao usuário
  audioAtual.addEventListener('loadstart', function() {
    showStatus(statusDiv, '📁 Carregando arquivo de áudio...', 'info');
    console.log('📁 Iniciando carregamento do áudio');
  });

  audioAtual.addEventListener('canplay', function() {
    showStatus(statusDiv, '✅ Áudio pronto! Reproduzindo...', 'success');
    console.log('✅ Áudio pronto para reprodução');
  });

  audioAtual.addEventListener('playing', function() {
    showStatus(statusDiv, '🔊 Reproduzindo explicação da Estela...', 'success');
    console.log('▶️ Áudio reproduzindo');
  });

  audioAtual.addEventListener('pause', function() {
    showStatus(statusDiv, '⏸️ Áudio pausado', 'info');
    console.log('⏸️ Áudio pausado');
  });

  audioAtual.addEventListener('ended', function() {
    showStatus(statusDiv, '✅ Explicação concluída!', 'success');
    console.log('🔴 Áudio finalizado');
  });
}

```

```

    audioAtual = null;
  });

  audioAtual.addEventListener('error', function(e) {
    const erro = e.target.error;
    let mensagem = '❌ Erro ao carregar áudio: ';

    switch(erro.code) {
      case 1: mensagem += 'Carregamento abortado.'; break;
      case 2: mensagem += 'Arquivo não encontrado. Verifique se o arquivo existe
na pasta "audios".'; break;
      case 3: mensagem += 'Erro de decodificação. Tente converter para MP3.';
break;
      case 4: mensagem += 'Formato não suportado. Use MP3, WAV ou OGG.';
break;
      default: mensagem += 'Erro desconhecido.'; break;
    }

    showStatus(statusDiv, mensagem + ' 📖 Texto disponível para leitura.', 'error');
    console.log('❌ Erro no áudio:', erro);
    console.log('🔍 Tentando carregar:', audioUrl);
  });

  // 5. Tentar reproduzir
  audioAtual.play().then(function() {
    console.log('🎵 Reprodução iniciada com sucesso');
  }).catch(function(error) {
    console.log('❌ Erro ao reproduzir:', error);

    if (error.name === 'NotAllowedError') {
      showStatus(statusDiv, '🔊 Clique no ícone 🔊 do navegador e permita áudio,
depois clique novamente no botão.', 'warning');
    } else if (error.name === 'NotSupportedError') {
      showStatus(statusDiv, '❌ Formato de áudio não suportado. Texto disponível
para leitura.', 'error');
    } else {
      showStatus(statusDiv, '❌ Erro na reprodução: ' + error.message + '. Texto
disponível para leitura.', 'error');
    }
  });

  } catch (error) {
    showStatus(statusDiv, '❌ Erro geral no sistema de áudio: ' + error.message,
'error');
    console.log('❌ Erro geral:', error);
  }
}

```

```
// ===== FUNÇÃO PARA MOSTRAR STATUS =====
function showStatus(element, message, type) {
  if (element) {
    element.textContent = message;
    element.className = 'status-audio show';

    // Remover classe anterior e adicionar nova
    element.classList.remove('info', 'success', 'warning', 'error');
    element.classList.add(type);

    // Auto-esconder após alguns segundos para mensagens de sucesso
    if (type === 'success') {
      setTimeout(() => {
        element.classList.remove('show');
      }, 5000);
    }
  }
}

// ===== CONTROLES DE ÁUDIO =====
function pararAudio() {
  if (audioAtual) {
    audioAtual.pause();
    audioAtual.currentTime = 0;
    audioAtual = null;
    console.log('🔴 Áudio parado');

    // Atualizar status
    const statusDiv = document.querySelector('.status-audio.show');
    if (statusDiv) {
      showStatus(statusDiv, '🔴 Áudio parado', 'info');
    }
  }
}

function pausarAudio() {
  if (audioAtual && !audioAtual.paused) {
    audioAtual.pause();
    console.log('⏸️ Áudio pausado');
  }
}

function continuarAudio() {
  if (audioAtual && audioAtual.paused) {
    audioAtual.play().then(() => {
      console.log('▶️ Áudio retomado');
    }).catch(error => {
      console.log('❌ Erro ao retomar áudio:', error);
    });
  }
}
```



```

    });
  }
}

```

```

function mudarVelocidade(velocidade) {
  velocidadeAtual = velocidade;
  if (audioAtual) {
    audioAtual.playbackRate = velocidade;
  }

  // Atualizar visual dos botões de velocidade
  document.querySelectorAll('.audio-controls button').forEach(btn => {
    btn.classList.remove('active');
  });
  event.target.classList.add('active');

  console.log('🎵 Velocidade alterada para:', velocidade + 'x');
}

```

```

// ===== NAVEGAÇÃO ENTRE AULAS =====
function irParaAula(numero) {
  console.log('📖 Navegando para aula:', numero);

  // Parar áudio se estiver tocando
  pararAudio();

  // Esconder todas as aulas e textos
  document.querySelectorAll('.aula').forEach(aula => {
    aula.classList.remove('active');
  });

  document.querySelectorAll('.texto-estela').forEach(texto => {
    texto.classList.remove('ativo');
  });

  document.querySelectorAll('.status-audio').forEach(status => {
    status.classList.remove('show');
  });

  // Mostrar aula selecionada
  const aulaElement = document.getElementById('aula' + numero);
  if (aulaElement) {
    aulaElement.classList.add('active');
    aulaAtual = numero;

    // Atualizar botões do menu
    document.querySelectorAll('.menu-btn').forEach(btn => {
      btn.classList.remove('active');
    });
  }
}

```

```

});
document.querySelectorAll('.menu-btn')[numero - 1].classList.add('active');

console.log('✅ Aula', numero, 'ativada');
}

// Rolar para o topo
window.scrollTo(0, 0);
}

function proximaAula() {
  if (aulaAtual < 8) {
    irParaAula(aulaAtual + 1);
  }
}

function aulaAnterior() {
  if (aulaAtual > 1) {
    irParaAula(aulaAtual - 1);
  }
}

function reiniciarCurso() {
  if (confirm('🔄 Tem certeza que deseja reiniciar o curso?')) {
    irParaAula(1);
  }
}

// ===== SISTEMA DE QUIZ =====
function responder(elemento, correto) {
  const opcoes = elemento.parentNode.querySelectorAll('.opcao');
  opcoes.forEach(opcao => {
    opcao.style.pointerEvents = 'none';
    opcao.classList.remove('correta', 'incorreta');
  });

  if (correto) {
    elemento.classList.add('correta');
  } else {
    elemento.classList.add('incorreta');
    // Marcar a resposta correta
    opcoes.forEach(opcao => {
      if (opcao.onclick && opcao.onclick.toString().includes('true')) {
        opcao.classList.add('correta');
      }
    });
  }
}

```

```

// ===== INICIALIZAÇÃO E DEBUG =====
console.log('🎮 Curso de Desenvolvimento de Jogos 2D carregado!');
console.log('🔧 Sistema de áudio configurado para GitHub Pages');
console.log('📊 Total de aulas:', Object.keys(audiosEstela).length);
console.log('🌐 Áudios funcionarão perfeitamente online!');
console.log('💡 Para testar, clique em "Ouvir explicação da Estela");

// ===== TESTE DE DEBUGGING =====
function testarAudio() {
  console.log('🔧 TESTE: Tentando carregar aula1.mp3...');
  const audio = new Audio(audiosEstela[1]);

  audio.addEventListener('loadstart', () => console.log('🟡 Iniciando carregamento...'));
  audio.addEventListener('canplay', () => console.log('🟢 Arquivo encontrado e pronto!'));
  audio.addEventListener('error', (e) => {
    console.log('🔴 ERRO DETALHADO:', e.target.error);
    console.log('🔴 Código do erro:', e.target.error.code);
    console.log('🔴 Tentando carregar:', e.target.src);
    console.log('💡 Verifique se o arquivo aula1.mp3 está na pasta audios/");
  });

  audio.load(); // Força o carregamento
}

// Executar teste após 3 segundos
setTimeout(testarAudio, 3000);
</script>
</body>
</html>

```