

# CCO A CRITICK



Faculdade Estácio de Sá.

Projeto de desenvolvimento Web, HTML, Java Script e PHP.

Alunos(as): Caio Vinícius, David Arthur, Ezequiel Barbosa, Ingrid Santana, Isabela Santana. Professor: Davi Barros



# Carca Critick

O Carca Critick é um site especializado na avaliação de jogos, com o objetivo de oferecer uma plataforma interativa e de fácil navegação para os usuários. Com avaliações e reviews honestos, poupamos tempo e recursos dos usuários. Em nossa plataforma, o usuário terá acesso a um universo de informações sobre os últimos lançamentos, jogos atuais e clássicos atemporais.



# Por que criar um site de críticas de jogos?

Crescimento do mercado de jogos: Considerando o crescimento contínuo da indústria de jogos, que em 2024 as expectativas de arrecadamento de dinheiro podem chegar a cerca US\$ 187,7 bilhões, 2,1% a mais que o arrecadado no ano de 2023.

Engajamento do público: De acordo com a Game Developer's Conference (GDC), Digital Game World, jogadores são altamente engajados e formam comunidades em torno de jogos, criando uma oportunidade para influenciar suas decisões de compra.

Críticas ajudam a escolher jogos com base na qualidade e nas preferências pessoais.

Monetização

Autonomia e credibilidade.

Interatividade e Feedback: Diálogo com a audiência.



# Desafios na criação para o mercado de jogos

Alta competição, tendo em vista que o mercado de críticas de jogos está saturado com grandes sites como IGM, Polygon e Youtube com influenciadores dominando a audiência.

Manter a qualidade, credibilidade e conteúdo atualizado. Engajamento com o público.

Acompanhamento de tendências tecnológicas.

Desafios técnicos e de design: Criar um site bem estruturado requer conhecimentos técnicos, recursos para manutenção e um planejamento cuidadoso da experiência do usuário.

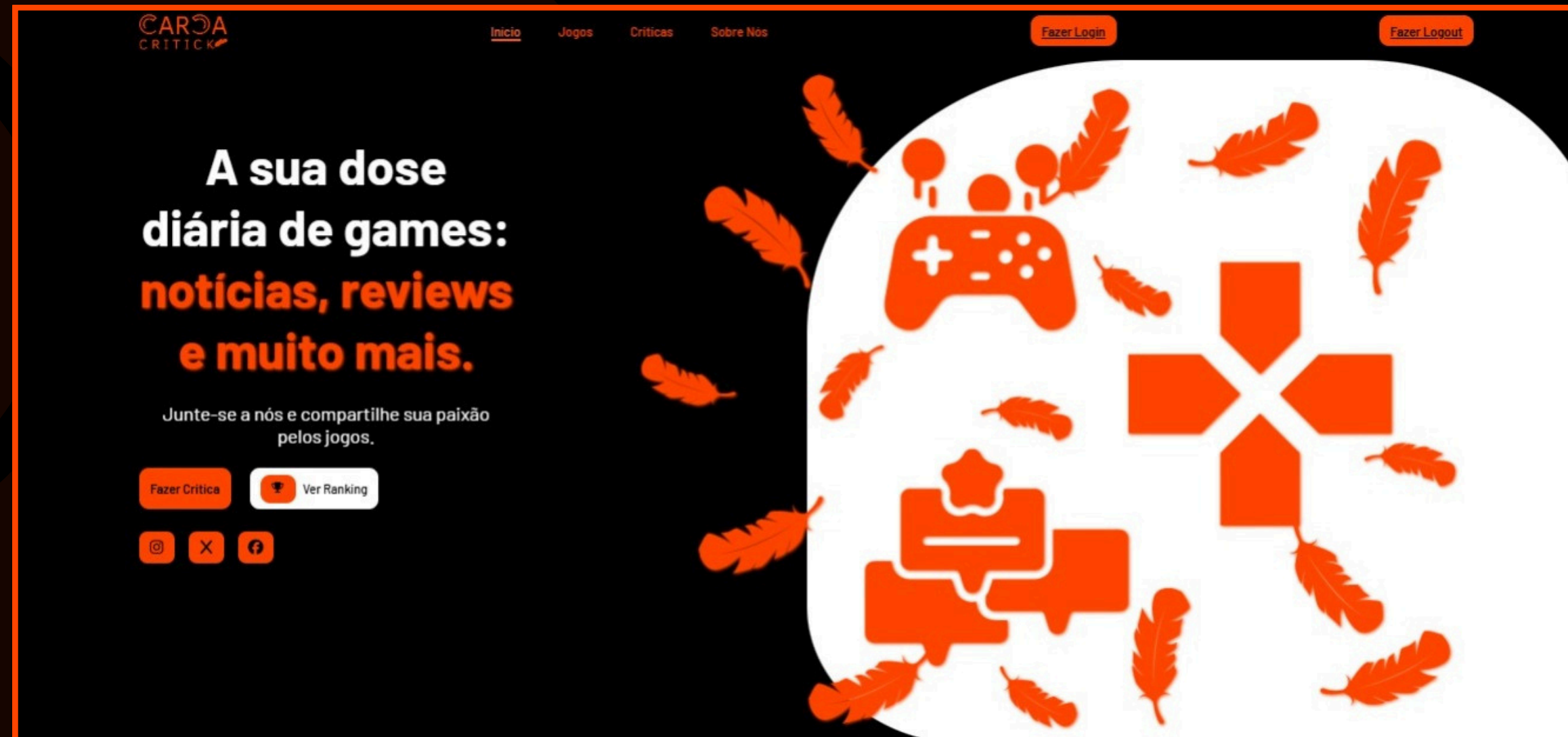


# Além da Superfície: Críticas e Análises Profundas

Para o futuro do nosso site, desejamos transmitir uma imagem de credibilidade, especialização e responsabilidade, oferecendo aos nossos usuários análises imparciais e detalhadas sobre jogos, enquanto compartilhamos nossa paixão pelo universo gamer. Além disso, buscaremos transmitir acessibilidade e uma experiência intuitiva, sempre atualizada.

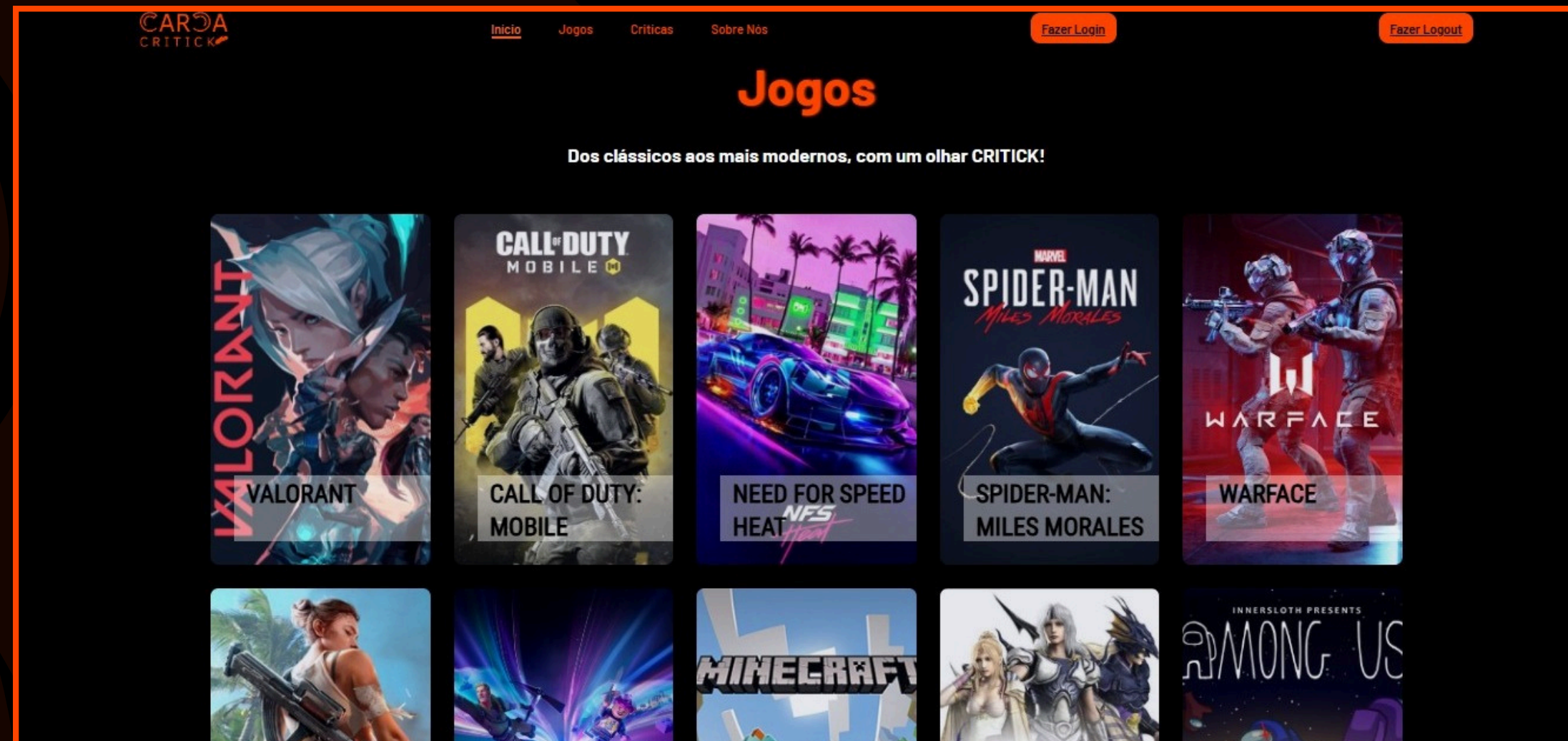


# Site: Produto final

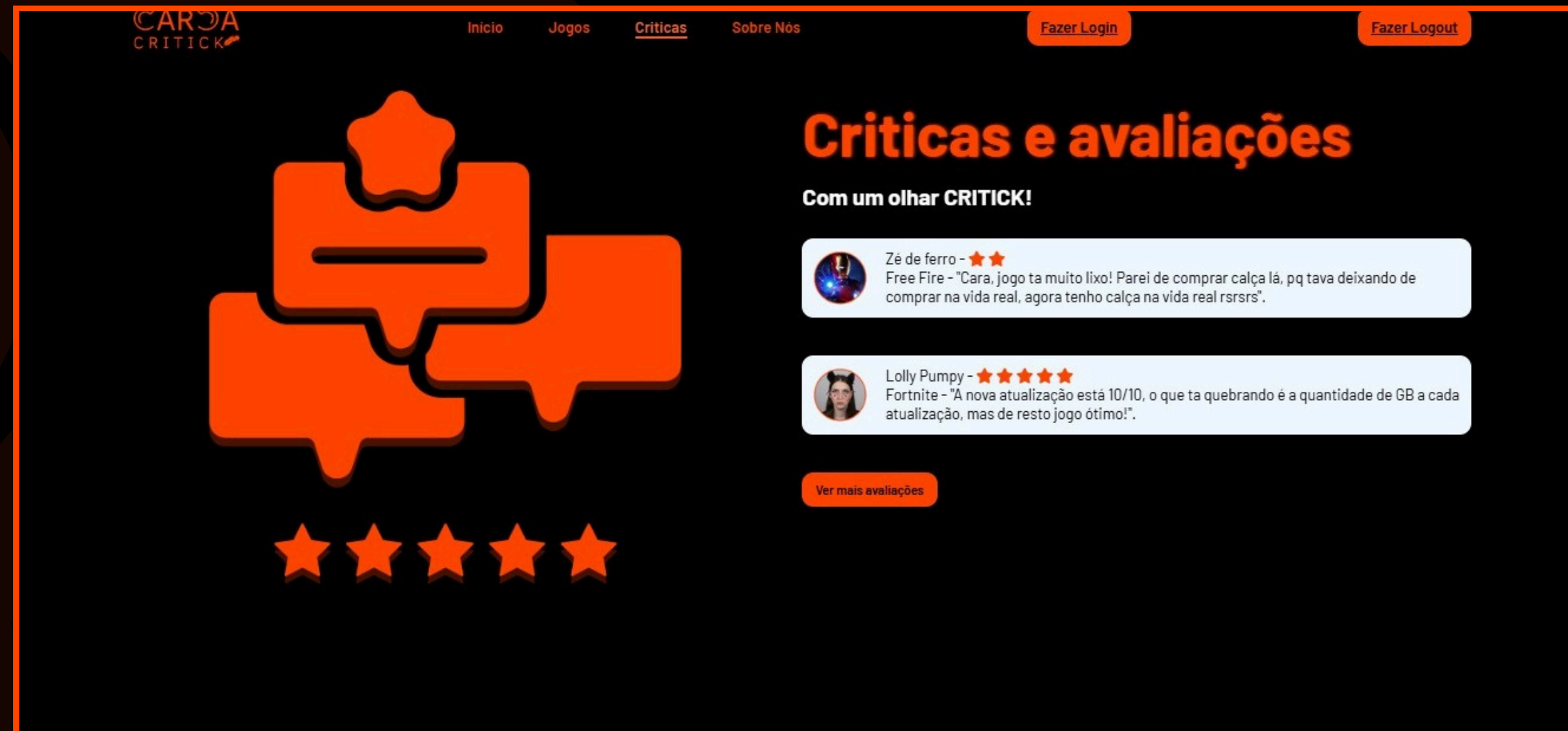




# Site: Produto final



# Site: Produto final





# Site: Produto final

CARDA  
CRITICK

[Início](#)[Jogos](#)[Críticas](#)[Sobre Nós](#)

Fazer Login

Top 10 Jogos

Posição	Jogo	Nota
1	Minecraft (2009)	5.0
2	Final Fantasy IV (1991)	4.8
3	Spider-Man: Miles Morales (2020)	4.6
4	Among Us (2018)	4.4
5	Valorant (2020)	4.2
6	Fortnite (2017)	3.9
7	Free Fire (2017)	3.7
8	Need for Speed Heat (2019)	3.6



# Código Back-end

## Formulário de Login

```
<!-- Começo do Formulario de Login -->
<form method="POST" action="login_query.php">
    <div class="alert alert-info">Login</div>
    <div class="form-group">
        <label>Usuario</label>
        <input type="text" name="usuario" class="form-control" required="required"/>
    </div>
    <div class="form-group">
        <label>Password</label>
        <input type="password" name="senha" class="form-control" required="required"/>
    </div>
    <?php
        //checando se a sessao 'error' esta estabelecida. Error session se a mensagem if the 'Username' and 'Password' is not valid.
        if(isset($_SESSION['error'])){
        ?>
        <!-- Mostrar mensagem Login error -->
        <div class="alert alert-danger"><?php echo $_SESSION['error']?></div>
        <?php
            //Desconfigurando a sessao 'error' após exibir a mensagem.
            unset($_SESSION['error']);
        }
        ?>
        <button class="btn btn-primary btn-block" name="login"><span class="glyphicon glyphicon-log-in"></span> Login</button>
    </form>
<!-- Fim do Formulario de Login -->
```



# Código Back-end

## Envio de Criticas

```
save_critica.php
1  <?php
2      //comecando a sessao
3      session_start();
4      //Incluindo a conexao com o banco de dados
5      require_once 'conn.php';
6
7      if(isset($_POST['critica'])){
8          // configurando variaveis
9          $usuario = $_POST['usuario'];
10         $jogo = $_POST['jogo'];
11         $nota = $_POST['nota'];
12         $comentario = $_POST['comentario'];
13         // Consulta para verificar se o usuário existe na tabela users
14         $queryVerificaUsuario = "SELECT COUNT(*) FROM users WHERE usuario = :usuario";
15         $stmtVerifica = $conn->prepare($queryVerificaUsuario);
16         $stmtVerifica->bindParam(':usuario', $usuario);
17         $stmtVerifica->execute();
18         // Verifica se o usuário foi encontrado
19         if ($stmtVerifica->fetchColumn() > 0) {
20             // Inserindo Consulta para envio da critica
21             $queryCritica = "INSERT INTO `critica` (usuario, jogo, nota, comentario) VALUES(:usuario, :jogo, :nota, :comentario)";
22             $stmt = $conn->prepare($queryCritica);
23             $stmt->bindParam(':usuario', $usuario);
24             $stmt->bindParam(':jogo', $jogo);
25             $stmt->bindParam(':nota', $nota);
26             $stmt->bindParam(':comentario', $comentario);
27             // Checando se a consulta foi um sucesso
28             if($stmt->execute()){
29                 //Configurando uma sessao 'sucesso' para salvar ou inserir a mensagem de sucesso.
30                 $_SESSION['sucesso'] = "Critica feita com sucesso";
31                 //redirecionando para critica.php
32                 header('location: critica.php');
33             }
34         } else {
35             // Usuário não encontrado, exibe mensagem de erro.
36             $_SESSION['erro'] = "Usuário não encontrado. Por favor, se registre e faça login.";
37             header('location: critica.php');
```



# Código Back-end

## Conexão do Banco de Dados

```
<?php
// Verificando se o arquivo do banco de dados existe. Se não, ele é criado vazio.
if (!is_file('db/db_users.sqlite3')) {
|   file_put_contents('db/db_users.sqlite3', null);
}
// Conectando ao banco de dados
$conn = new PDO('sqlite:db/db_users.sqlite3');
// Configurando os atributos da conexão
$conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
// Consulta para criar a tabela de users no banco de dados, se ela ainda não existir.
$query = "CREATE TABLE IF NOT EXISTS users(user_id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, usuario TEXT, senha TEXT)";
// Executando a consulta para criar a tabela de usuários
$conn->exec($query);
// Consulta para criar a tabela critica no banco de dados, se ela ainda não existir.
$queryCritica = "CREATE TABLE IF NOT EXISTS critica(usuario TEXT NOT NULL, jogo TEXT NOT NULL, nota INTEGER NOT NULL CHECK (nota >= 1 AND nota <= 5)";
// Executando a consulta para criar a tabela de críticas
$conn->exec($queryCritica);
?>
```



# Código Front-end

Este trecho de código HTML serve para incluir a biblioteca jQuery em uma página da web.

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.5.1/jquery.min.js"  
integrity="sha512-bLT0Qm9VnAYZDflyKcBaQ2gg0hSYNQrJ8RilYldYQ1FxQYoCLtUjuuRuZo+  
fjqhx/qtq/1itJ0C2ejDxltZVFg=="  
crossorigin="anonymous" referrerpolicy="no-referrer">  
</script>
```





# Código Front-end

## Barra de Navegação - Navbar

Código no HTML:

```
<nav id="navbar">
  <div class="logo"> </div>
  <ul id="nav_list">

    <li class="nav-item active">
      <a href="#home">Início</a>
    </li>

    <li class="nav-item">
      <a href="#games">Jogos</a>
    </li>

    <li class="nav-item">
      <a href="#critick">Criticas</a>
    </li>

    <li class="nav-item">
      <a href="#abautus">Sobre Nós</a>
    </li>

  </ul>
```

Código no CSS:

```
header {
  width: 100%;
  padding: 10px 8%;
  position: sticky;
  top: 0;
  background-color: rgba(0, 0, 0, 0.616);
  z-index: 3;
}

#navbar {
  width: 100%;
  display: flex;
  align-items: center;
  justify-content: space-between;
}

#nav_Logo {
  font-size: 24px;
  color: #FF4500;
}

#nav_List {
  display: flex;
  list-style: none;
  gap: 48px;
}
```



# Código Front-end

Este código CSS é usado para criar uma versão simplificada do layout para telas menores:

```
@media screen and (max-width: 1170px) {  
  #nav_list,  
  #navbar .btn-default {  
    display: none;  
  }  
}
```

Seletor : Hover no CSS.

```
.btn-default:hover {  
  background-color: #FF4500;  
  box-shadow: 0px 0px 12px 4px #FF4500;  
}
```



# Código Front-end

O código completo cria um link que, quando clicado, levará o usuário para a página "pag011.html", no caso uma página iframe.

```
<div class="container10">
  <div class="card">
    
    <div class="card-content">
      <h2>Among Us&ltinput type="text"/></h2>
      <p>Among Us é um jogo onde um grupo de tripulantes precisa completar tarefas em uma nave espacial.</p>
      <a href="pag011.html">Veja mais &ltspan class="material-symbols-outlined">arrow_right_alt</span></a>
      <!--No codigo HTML a cima foi criado uma especie de link de naveção interna, para melhor experiênciado usuário-->
    </div>
  </div>
</div>
</div>
```

## Código Front-end

Essa linha de código JavaScript utiliza a biblioteca ScrollReveal.js para criar uma animação em um elemento HTML com o ID "cta" (provavelmente um call to action ou botão de chamada para ação), que pode ser observada ao rolar a página para baixo.

```
ScrollReveal().reveal('#cta', {  
  origin: 'left',  
  duration: 2000,  
  distance: '20%'  
})
```



# Carca Critick

Agradecemos a todos pela atenção e por acompanharem nossa apresentação.  
Foi um prazer compartilhar o projeto com vocês.  
Muito obrigado!





## Referências bibliográficas:

<https://digital.sebraers.com.br/blog/mercado/tendencias-para-a-industria-de-games-em-2024/>

<https://valor.globo.com/empresas/noticia/2024/08/13/faturamento-do-setor-de-jogos-eletronicos-deve-crescer-21percent-em-2024-mesmo-com-incertezas.ghtml>

<https://gdconf.com/> <https://warpzone.me/o-game-design-e-seus-desafios-no-desenvolvimento-de-games/>

