

Apresentação do Site da Roleta De Desafios.

O site foi desenvolvido para proporcionar uma experiência interativa e envolvente para os usuários que

desejam testar sua sorte de maneira divertida. Com uma interface dinâmica e intuitiva, a plataforma permite que os visitantes participem de desafios exclusivos.

1. Propósito do Site: O objetivo do site é oferecer entretenimento por meio de um sistema de roleta digital, onde os usuários podem girar a roda.

2. Estrutura e Navegação: A navegação no site é simples e acessível, garantindo uma experiência fluida para todos os públicos. Entre os principais recursos estão:

- Página inicial com destaque para as rodadas disponíveis.
- Roleta para o sorteio dos desafios
- Jogos aleatórios ou da sua escolha

3. Design e Identidade Visual: O design do site foi elaborado para transmitir diversão e emoção. Utilizando cores vibrantes, efeitos animados e elementos gráficos dinâmicos, a plataforma proporciona uma atmosfera envolvente que incentiva a interação dos usuários.

4. Funcionalidades Interativas: Para tornar a experiência ainda mais imersiva, o site conta com diversas funcionalidades, como:

- Roleta De Desafios animada com efeitos visuais
- Roleta de Desafios.
- Os usuários podem escolher por categorias que são 4:

1-Físico

2-Criativo

3-Engraçado

4-Aleatório

5. Otimização e Acessibilidade: O site foi projetado para ser totalmente responsivo, garantindo compatibilidade com diferentes dispositivos, desde desktops até smartphones.

6. Divulgação e Crescimento: O site da Roleta De Desafios combina tecnologia e diversão, proporcionando aos usuários uma experiência empolgante e interativa, com desafios emocionantes. Sempre procurando crescer no mundo dos sites de entretenimento.

--IMAGENS DO PROTOTIPO INICIAL DESENVOLVIDO



O HTML define a estrutura do site, organizando os elementos visíveis na tela.

Código Importante do HTML;

<header>

</header>

Explicação:

- Cria um cabeçalho (<header>) com a logo do site.
- A imagem da logo é carregada a partir do arquivo "logo.png".

<main class="container">

```

<section class="cards-container">
<article class="card">
<h2>ALEATÓRIO</h2>
<section class="card-image">

</section>
<button id="botaoAleatorio">JOGAR</button>
</article>
<article class="card">
<h2>CATEGORIA</h2>
<section class="card-image">

</section>
<button id="botaoCategoria">JOGAR</button>
<aside id="categoriasContainer" class="categorias-container"></aside>
</article>
</section>
</main>

```

Explicação:

Contém dois cards principais:

- “Aleatório” → Gira a roleta e sorteia um desafio aleatório.
- “Categoria” → Permite escolher um tipo de desafio antes de girar a roleta.

Cada card tem:

- Uma imagem ().
- Um botão “Jogar” (<button>).

```

<section class="roleta-container">
<aside class="ponteiro"></aside>
<article class="roleta" id="roleta"></article>
<p class="resultado" id="resultado"></p>
</section>

```

Explicação:

- Área onde a roleta aparece após o usuário clicar no botão “Jogar”.
- O <aside class="ponteiro"> representa a seta fixa que aponta para o resultado da roleta.

```
<section class="box-roleta-legenda">
  <article class="roleta-container">
    <aside class="ponteiro"></aside>
    <article class="roleta" id="roleta"></article>
    <p class="resultado" id="resultado"></p>
  </article>
```

```
<section class="legenda-roleta">
  <h3>Legenda de Cores da Roleta</h3>
  <ul>
    <li><span class="cor legenda-vermelho"></span> Físico</li>
    <li><span class="cor legenda-azul"></span> Engraçado</li>
    <li><span class="cor legenda-verde"></span> Inteligente</li>
    <li><span class="cor legenda-amarelo"></span> Físico</li>
    <li><span class="cor legenda-roxo"></span> Engraçado</li>
    <li><span class="cor legenda-laranja"></span> Inteligente</li>
  </ul>
</section>
</section>
```

Explicação:

- Contém a roleta (.roleta-container) com ponteiro, resultado e a legenda de cores abaixo dela.

CSS - Estilização do Site

O CSS define a aparência do site, incluindo cores, tamanhos e animações.

Estilização do Corpo e Cabeçalho

```
body {
  background-color: #002855;
  color: white;
```

```
text-align: center;
}
header {
background-color: #d6801a;
padding: 0;
}
header img {
height: 100px;
width: 500px;
}
```

Explicação:

- Define o fundo azul escuro do site.
- A cor laranja no cabeçalho (background-color: #d6801a).
- Ajusta o tamanho da logo (width: 500px, height: 100px).

Estilização dos Cards

```
.cards-container {
display: flex;
gap: 20px;
width: auto;
}
.card {
background-color: #d6801a;
padding: 20px;
border-radius: 10px;
text-align: center;
width: 220px;
height: 320px;
display: flex;
flex-direction: column;
align-items: center;
justify-content: space-between;
```

```
}
```

Explicação:

- Os cards são organizados lado a lado com `display: flex`.
- A cor laranja e as bordas arredondadas (`border-radius: 10px`) dão um efeito visual moderno.

Estilização da Roleta

```
.roleta-container {  
  position: relative;  
  
  width: 300px;  
  
  height: 380px;  
  
  display: flex;  
  
  flex-direction: column;  
  
  align-items: center;  
  
  justify-content: space-between;  
}
```

Explicação:

- Centraliza a roleta e define um tamanho fixo de 300px x 380px.

```
.roleta {  
  width: 300px;  
  
  height: 300px;  
  
  border-radius: 50%;  
  
  background: conic-gradient(  
    red 0deg 60deg,  
    blue 60deg 120deg,  
    green 120deg 180deg,  
    yellow 180deg 240deg,  
    purple 240deg 300deg,  
    orange 300deg 360deg  
  );  
  
  transition: transform 4s ease-out;  
  
  transform-origin: center;  
}
```

Explicação:

- Usa `conic-gradient()` para criar um efeito visual de setores coloridos.

A roleta gira suavemente devido ao `transition: transform 4s ease-out`.

JavaScript - Lógica do Jogo

O JavaScript faz a roleta girar e exibe o desafio sorteado.

Lista de Desafios

- Lista de desafios para cada categoria

const desafios = {

"Físico": [

"Fazer 10 polichinelos",

"Pular corda por 30 segundos",

"Ficar em prancha por 20 segundos",

"Correr no lugar por 1 minuto",

"Fazer 5 flexões",

"Pular em um pé só por 30 segundos",

"Fazer 10 agachamentos",

"Equilibrar-se com um pé só por 15 segundos",

"Fingir que está nadando por 20 segundos",

"Imitar um robô dançando",

"Fazer uma corrida até a parede e voltar"

],

"Engraçado": [

"Contar uma piada sem rir",

"Fazer uma careta",

"Falar como um robô por 1 minuto",

"Imitar um animal",

"Fingir que está em câmera lenta",

"Fazer uma dança engraçada",

"Inventar uma palavra e dar um significado para ela",

"Imitar uma celebridade ou personagem",

"Dançar sem música por 20 segundos",

```

    "Fingir que está chorando dramaticamente",
    "Fingir que é um apresentador de TV"
  ],
  "Inteligente": [
    "Resolver um enigma",
    "Dizer o alfabeto de trás para frente",
    "Fazer uma conta de cabeça difícil",
    "Falar 5 capitais do mundo",
    "Criar uma rima com a palavra 'foco'",
    "Fazer um trava-língua 3 vezes seguidas",
    "Dizer o nome de 3 filmes com a letra B",
    "Falar 5 palavras que rimam com 'bola'",
    "Resolver uma conta difícil que o grupo determinar",
    "Falar o maior número de animais que conseguir em 15 segundos"
  ]
};

```

Define quais cores representam cada categoria na roleta

```

const coresPorCategoria = {
  "Físico": "Vermelho / Amarelo",
  "Engraçado": "Azul / Roxo",
  "Inteligente": "Verde / Laranja"
};

```

Explicação:

- Define os desafios organizados por categorias: Físico, Engraçado e Inteligente.

Exibir Botões de Categoria ao Clicar em “Escolher Categoria”

```

document.getElementById("botaoCategoria").addEventListener("click", function () {
  let container = document.getElementById("categoriasContainer");
  container.innerHTML = ""; // Limpa categorias anteriores
  container.style.display = "flex"; // Exibe os botões em linha

```



```

Object.keys(desafios).forEach(categoria => {

    let botao = document.createElement("button");

    botao.innerText = categoria;

    botao.onclick = () => iniciarRoletaCategoria(categoria); // Define ação do botão

    container.appendChild(botao);

});
});

```

Explicação:

Quando o jogador clica em “Escolher Categoria”, botões são criados dinamicamente para cada categoria de desafio.

Função para Escolher Desafio Aleatoriamente

```

function escolherDesafio(categoria) {

    let lista = desafios[categoria];

    return lista[Math.floor(Math.random() * lista.length)];

}

```

Explicação:

Sorteia um desafio aleatório da lista da categoria selecionada.

Iniciar Roleta com Categoria Escolhida

```

function iniciarRoletaCategoria(categoriaSelecionada) {

    girarRoleta(() => {

        const desafio = escolherDesafio(categoriaSelecionada);

        document.getElementById("resultado").innerText = `Categoria: ${categoriaSelecionada}
Cor: ${coresPorCategoria[categoriaSelecionada]}
Desafio: ${desafio}`;

    });

}

```

Explicação:

Gira a roleta apenas para efeito visual. O desafio é escolhido da categoria clicada pelo jogador.

Botão de Roleta Aleatória

```
document.getElementById("botaoAleatorio").addEventListener("click", () => {  
    iniciarRoletaAleatoria();  
});
```

Iniciar Roleta com Categoria Sorteada

```
function iniciarRoletaAleatoria() {  
    girarRoleta((categoriaSorteada) => {  
        const desafio = escolherDesafio(categoriaSorteada);  
        document.getElementById("resultado").innerText = `Categoria: ${categoriaSorteada}  
Cor: ${coresPorCategoria[categoriaSorteada]}  
Desafio: ${desafio}`;  
    });  
}
```

Explicação:

A roleta gira, sorteia a categoria com base no ângulo final e mostra um desafio aleatório correspondente.

Função para Girar a Roleta

```
let anguloAtual = 0; // Guarda o estado da rotação atual
```

```
function girarRoleta(callback) {  
    const roleta = document.getElementById("roleta");  
    const novaRotacao = 3600 + Math.random() * 360; // Gira entre 3600 e 3960 graus  
    anguloAtual += novaRotacao;  
  
    const anguloFinal = anguloAtual % 360;
```

```

// Reset visual antes da animação

roleta.style.transition = "none";

roleta.style.transform = `rotate(${anguloFinal}deg)`;


// Aplica animação

setTimeout(() => {

    roleta.style.transition = "transform 4s cubic-bezier(0.17, 0.67, 0.83, 0.67)";

    roleta.style.transform = `rotate(${anguloAtual}deg)`;

}, 50);


// Após animação (4,5 segundos), chama callback com categoria

setTimeout(() => {

    const categoria = categoriaPorAngulo(anguloFinal);

    if (callback) callback(categoria);

}, 4500);

}

```

Explicação:

Controla a rotação animada da roleta. Quando termina, o jogo calcula em qual ângulo parou e retorna a categoria correspondente.

Definir Categoria Baseada no Ângulo da Roleta

```

function categoriaPorAngulo(angulo) {

    angulo = angulo % 360;


    if (angulo >= 0 && angulo <= 60) return "Inteligente";

    if (angulo > 60 && angulo <= 120) return "Engraçado";

    if (angulo > 120 && angulo <= 180) return "Físico";

    if (angulo > 180 && angulo <= 240) return "Inteligente";

    if (angulo > 240 && angulo <= 300) return "Engraçado";

```

```
if (angulo > 300 && angulo < 360) return "Físico";
```

```
return "Físico"; // Valor padrão
```

```
}
```

Explicação:

Divide o círculo da roleta em 6 fatias de 60°. Cada fatia representa uma categoria. Assim, com base no ângulo final da rotação, é possível determinar qual categoria foi sorteada. • A roleta gira e para em um ângulo aleatório.

- O sistema divide a roleta em 6 fatias (60° cada).
- Com base no ângulo final, ele determina a categoria sorteada.
- Um desafio aleatório da categoria é então escolhido.
- O resultado com a categoria, cor e desafio aparece na tela.

Resumo Final

HTML → Estrutura com cards, roleta e botões.

CSS → Estilização, efeitos visuais e animação da roleta.

JAVASCRIPT → Faz a roleta girar e sorteia desafios aleatórios.

----CONCLUSÃO COM OS APRENDIZADOS ADQUIRIDOS:

Este código cria uma estrutura para um site de entretenimento que oferece desafios aleatórios por meio de uma roleta. As funcionalidades principais são:

- Menu de navegação.
- Seção principal com uma roleta de desafios.
- Duas opções de jogo (Aleatório e Categoria) com botões de "Jogar".

Nome e RA:

- Pedro Oficiati-10723732
- Gustavo Frainer-10386596
- Melissa Rodrigues-10726749