**3D COLÉGIOS**

**JOÃO PEDRO**

**NICOLAS**

**ADVINHAÇÃO DE EMOJI**

**NOVA IGUAÇU**

**2024**

**JOÃO PEDRO**

NICOLAS

**ADVINHAÇÃO DE EMOJI**

Trabalho acadêmico apresentado à disciplina de LTP como requisito de nota parcial para o segundo semestre da Turma RGI42. Requerido pelo prof. Gabriel

**NOVA IGUAÇU**

**2024**

**SUMÁRIO**

[1 INTRODUÇÃO 4](#_Toc183469342)

[2 DESENVOLVIMENTO 5](#_Toc183469343)

[3 CONSIDERAÇÕES FINAIS 10](#_Toc183469344)

[REFERÊNCIAS 11](#_Toc183469345)

# 1 INTRODUÇÃO

Este projeto consiste no desenvolvimento de um site interativo para um jogo de adivinhação de emojis, utilizando as linguagens PHP, HTML e CSS. O objetivo principal é criar uma aplicação dinâmica e divertida, que proporcione ao usuário uma experiência leve e intuitiva. Além de aplicar os conhecimentos adquiridos no curso técnico de informática, o trabalho busca explorar práticas de programação web, promovendo criatividade e aprendizado.

# 2 DESENVOLVIMENTO

Início do Projeto

Para iniciar o projeto de adivinhação de emojis, você precisará configurar um ambiente de desenvolvimento que permita executar código PHP localmente e acessar o jogo por meio de um navegador. Aqui estão as etapas iniciais para configurar o projeto:

1. Configuração do Servidor Local: O PHP é uma linguagem server-side, o que significa que você precisará de um servidor local para rodar o código PHP. Você pode usar ferramentas como XAMPP, WAMP ou MAMP para criar um servidor local em seu computador
2. Editor de Código: Utilize um editor de código para escrever os arquivos do seu projeto. Recomendamos editores como Visual Studio Code, Sublime Text ou Atom, que oferecem suporte a PHP, HTML e CSS com recursos como destaque de sintaxe e autocompletar.
3. Estrutura de Arquivos: Organize os arquivos do projeto da seguinte maneira

Index.php: O arquivo principal que contém a lógica do jogo e a estrutura HTML.

Style.css: Arquivo de estilo que define o design visual do site

Recursos Técnicos Utilizados

O projeto de adivinhação de emojis foi desenvolvido utilizando as seguintes tecnologias:

PHP (Hypertext Preprocessor): A principal linguagem de programação server-side utilizada para gerenciar a lógica do jogo. PHP é responsável por processar as respostas do jogador, verificar se estão corretas e gerar aleatoriamente os emojis para o jogo. Além disso, PHP pode ser usado para gerenciar sessões e armazenar informações temporárias, como a pontuação do jogador.

HTML (HyperText Markup Language): Usado para estruturar o conteúdo do site. No jogo de adivinhação, o HTML define a estrutura das páginas, como a exibição dos emojis, áreas para o jogador inserir suas respostas, e botões de interação. O HTML é essencial para criar a interface e garantir que o conteúdo seja organizado de forma lógica.

CSS (Cascading Style Sheets): CSS é responsável pela estilização visual do site. Ele define a aparência do jogo, como cores de fundo, tamanho e disposição dos emojis, tipografia e os estilos dos botões e caixas de respostas. Além disso, CSS pode ser usado para criar animações e transições suaves, tornando a experiência do usuário mais agradável.

Essas tecnologias trabalham juntas para criar uma experiência interativa e dinâmica, onde o PHP gerencia a lógica e o backend, o HTML estrutura o conteúdo e a interface, e o CSS cuida da estética e usabilidade do jogo.

Funcionamento do Jogo de adivinhação de Emojis:

O jogo de adivinhação de emojis utiliza PHP para gerenciar a lógica do backend, HTML para estruturar o conteúdo da página e CSS para estilizar o site.

1. Sessões em PHP:

O código começa com session\_start(), que ativa o uso de sessões no PHP. As sessões permitem armazenar dados que persistem entre requisições, como as vidas do jogador e os emojis enviados.

1. Recebendo a Resposta do Jogador:

Quando o jogador envia uma resposta pelo formulário, o PHP processa o valor enviado em $\_POST[‘valore’].

O valor da resposta é convertido para maiúsculas (strtoupper) e tem espaços extras removidos (trim) para garantir que não haja problemas de formatação.

A resposta é então armazenada na sessão em um array chamado $\_SESSION[‘emojis’].

1. Gerenciando as Vidas do Jogador:

A variável $vida controla o número de vidas do jogador. Se o jogador errar, as vidas são decrementadas, e o valor atualizado é salvo na sessão ($\_SESSION[‘vida’]).

Quando as vidas chegam a 0, o jogo termina e a sessão é destruída com session\_destroy().

1. Banco de Desafios

O array $traducao armazena os desafios e suas respectivas traduções em emojis. Cada chave do array é o nome de um anime, e o valor é sua representação com emojis.

Um anime é selecionado aleatoriamente usando a função rand() e armazenado na sessão como $\_SESSION[‘desafio’].

1. Lógica do Jogo:

O emoji correspondente ao desafio atual é exibido na página com <?php echo $desafio; ?>.

Quando o jogador envia uma resposta:

Se a resposta corresponder à chave correta no array $traducao (por exemplo, “NARUTO” para “👱‍♂️🍜😁🦊”), o jogador vence, e o jogo gera um novo desafio.

Se a resposta estiver errada, o jogador perde uma vida. Quando as vidas chegam a zero, o jogo exibe o nome correto do anime e termina.

1. Exibição do Resultado:

Mensagens de feedback são exibidas ao jogador:

“🎉 Você acertou! Parabéns!” para respostas corretas.

“❌ Errou! Você ainda tem X vidas.” Para respostas incorretas, onde X é o número de vidas restantes.

“💀 Você perdeu! O anime correto era: XYZ” quando o jogador perde todas as vidas, mostrando a resposta correta.

1. Reinício do Jogo:

Quando o jogador acerta, a sessão é atualizada com um novo desafio (unset($\_SESSION[‘desafio’])), e as vidas são restauradas para 3 ($\_SESSION[‘vida’] = 3).

# 3 CONSIDERAÇÕES FINAIS

O desenvolvimento deste site foi uma oportunidade de consolidar os conceitos aprendidos em PHP, HTML e CSS, permitindo a criação de uma aplicação funcional e atrativa. O jogo de adivinhação de emojis demonstra como a programação pode ser aplicada para entreter e engajar os usuários. Este projeto não apenas reforçou habilidades técnicas, mas também mostrou o potencial da tecnologia como ferramenta criativa e prática..

# REFERÊNCIAS