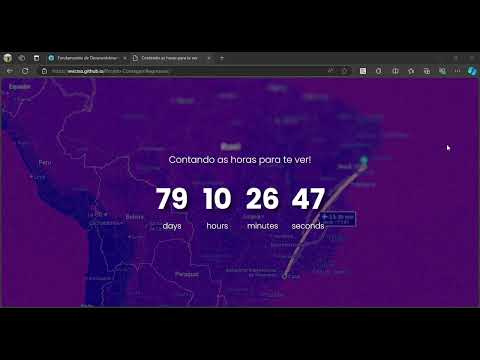
Link do Youtube: <https://youtu.be/SuR6wkmfTdU>

[](https://youtu.be/SuR6wkmfTdU)

Link do GitHub: [evicsss/Projeto-ContagemRegressiva (github.com)](https://github.com/evicsss/Projeto-ContagemRegressiva)

# **Documentação do Projeto: Contagem Regressiva**

## **Introdução**

Neste projeto, desenvolvi uma aplicação interativa de contagem regressiva utilizando JavaScript moderno. O objetivo foi criar uma ferramenta simples e visualmente atraente que permite ao usuário acompanhar a contagem do tempo até um evento específico.

## **Funcionalidades Implementadas**

### **1. Contagem Regressiva**

A principal funcionalidade da aplicação é a contagem regressiva para uma data alvo específica. O tempo restante é atualizado a cada segundo, mostrando dias, horas, minutos e segundos.

### **2. Interface Intuitiva**

A interface foi projetada para ser clara e fácil de entender, apresentando a contagem em quatro seções distintas (dias, horas, minutos e segundos) com uma mensagem central que indica o propósito da contagem.

### **3. Estilos Visuais**

Utilizei CSS para criar uma aparência atraente e responsiva. A aplicação conta com um fundo estilizado e textos que se destacam, proporcionando uma experiência visual agradável.

## **Processo de Desenvolvimento**

O desenvolvimento da aplicação seguiu as etapas abaixo:

1. **Planejamento**: Defini as funcionalidades principais e a estrutura da interface.
2. **Configuração do Ambiente**: Criei os arquivos necessários (HTML, CSS e JavaScript) e configurei o ambiente de desenvolvimento.
3. **Implementação da Lógica**:
   1. Utilizei setInterval para atualizar a contagem a cada segundo.
   2. Criei a função render para atualizar a interface com o tempo restante.
   3. Usei const para declarar variáveis imutáveis e objetos para encapsular os dados da contagem.
4. **Estilização**: Apliquei CSS para garantir que a aplicação fosse responsiva e visualmente agradável.
5. **Testes**: Realizei testes em diferentes navegadores e dispositivos para garantir o funcionamento adequado da contagem regressiva.

## **Instruções para Uso**

1. **Clonagem do Repositório**:
   1. Clone o repositório do GitHub: git clone <https://github.com/evicsss/Projeto-ContagemRegressiva.git>.
2. **Abertura do Arquivo HTML**:
   1. Navegue até a pasta do projeto e abra o arquivo index.html em um navegador.
3. **Interação com a Aplicação**:
   1. A contagem regressiva começará automaticamente, mostrando o tempo restante até a data especificada.
   2. A interface é atualizada a cada segundo até que o tempo expire.

## **Considerações Finais**

Este projeto não só me permitiu consolidar meus conhecimentos em JavaScript moderno, como também proporcionou uma experiência prática em desenvolvimento web.

Código JS:

const daysElement = document.querySelector("[data-days]");

const hoursElement = document.querySelector("[data-hours]");

const minutesElement = document.querySelector("[data-minutes]");

const secondsElement = document.querySelector("[data-seconds]");

const render = (days, hours, minutes, seconds) => {

daysElement.innerHTML = String(days).padStart("2", 0);

hoursElement.innerHTML = String(hours).padStart("2", 0);

minutesElement.innerHTML = String(minutes).padStart("2", 0);

secondsElement.innerHTML = String(seconds).padStart("2", 0);

};

const countdown = () => {

const now = new Date();

const targetDate = new Date(2024, 11, 7, 7, 55);

const timeLeft = targetDate - now;

const timeLeftObj = {

days: Math.floor(timeLeft / (1000 \* 60 \* 60 \* 24)),

hours: Math.floor((timeLeft % (1000 \* 60 \* 60 \* 24)) / (1000 \* 60 \* 60)),

minutes: Math.floor((timeLeft % (1000 \* 60 \* 60)) / (1000 \* 60)),

seconds: Math.floor((timeLeft % (1000 \* 60)) / 1000),

};

timeLeft <= 0

? clearInterval(interval)

: render(timeLeftObj.days, timeLeftObj.hours, timeLeftObj.minutes, timeLeftObj.seconds);

};

const interval = setInterval(() => countdown(), 1000);

countdown();

Código HTML:

<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Contando as horas para te ver</title>

<link rel="stylesheet" href="styles.css" />

</head>

<body>

<div class="countdown">

<span class="title">Contando as horas para te ver!</span>

<div class="box">

<span class="value" data-days>00</span>

<span class="name">days</span>

</div>

<div class="box">

<span class="value" data-hours>00</span>

<span class="name">hours</span>

</div>

<div class="box">

<span class="value" data-minutes>00</span>

<span class="name">minutes</span>

</div>

<div class="box">

<span class="value" data-seconds>00</span>

<span class="name">seconds</span>

</div>

</div>

<script src="app.js" defer></script>

</body>

</html>

Código CSS:

@import url("https://fonts.googleapis.com/css2?family=Poppins:wght@300;700&display=swap");

\* {

padding: 0;

margin: 0;

box-sizing: border-box;

font-family: "Poppins";

}

body {

display: grid;

place-content: center;

min-height: 100vh;

width: 100%;

background-image: url('img/map.png');

background-size: cover;

background-position: center;

background-repeat: no-repeat;

}

.countdown {

display: grid;

grid-template-columns: repeat(4, 1fr);

color: rgb(255, 255, 255);

gap: 32px;

}

.title {

grid-column: 4 span;

text-align: center;

font-size: 26px;

font-weight: 400;

text-shadow: 0.1em 0.1em 0.2em rgb(31, 1, 46)

}

.box {

display: grid;

grid-template-columns: 1fr;

text-align: center;

}

.value {

font-size: 80px;

font-weight: 700;

text-shadow: 0.1em 0.1em 0.1em rgb(31, 1, 46)

}

.name {

font-size: 22px;

text-shadow: 0.1em 0.1em 0.1em rgb(31, 1, 46)

}