

Svelte: mais um framework frontend?

André Lins



é hype que fala né



Agenda

- O que é Svelte?
- Conceitos básicos de svelte.
- Svelte Store
- Usando svelte dentro de outros frameworks.
- Sapper
- Revelt
- Comunidade Svelte



O que é Svelte?



Svelte é uma estrutura de componentes que compila seu código em javascript. Utilizando basicamente conceitos de html, css e javascript

Logo....



Svelte não é um framework



História do svelte

- Criado em 2016
- Autor: Rich Harris (criador do Rollup)
- v2: Desaparecimento de um framework
- v3: Repensando a reatividade
- Versão atual: 3.x



Características principais

- Não utiliza o Virtual DOM
- Totalmente reativo
- Tamanho: 3.6KB
- Sintaxe simples
- Baixa curva de aprendizado
- Biblioteca extensa para animações
- Novidade: **Suporte a Typescript**

★ 35,5K

👥 310

🕒 5,845

🔗 1,7K

Família Svelte



SVELTE

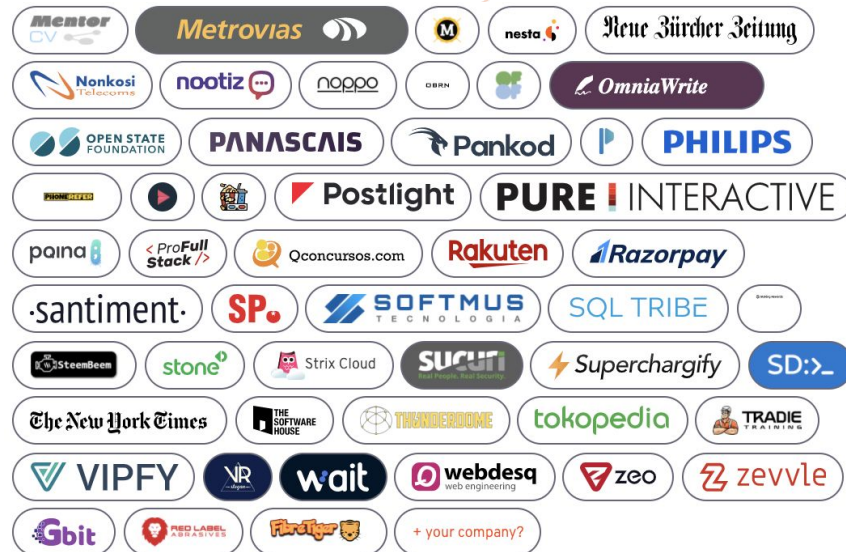


SAPPER

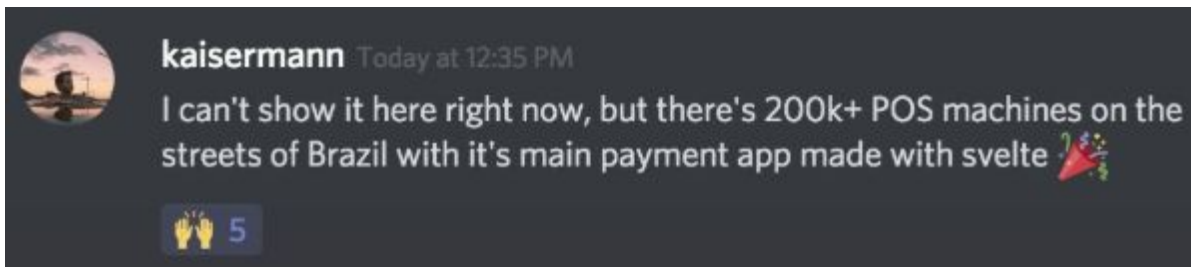


SVELTE
NATIVE

Empresas que utilizam



Case da Stone



Cadê esse código mágico?



Como iniciar um projeto svelte?



```
npx degit sveltejs/template my-svelte-project
```

```
# or download and extract
```

```
cd my-svelte-project
```

```
npm install
```

```
npm run dev
```

Hello World



```
<script>  
  let name = 'world';  
</script>  
  
<style>  
  h1 {  
    color: red;  
  }  
</style>  
  
<h1>Hello {name}!</h1>
```

Estruturas lógicas

```

<script>
  export let isLoggedIn;
</script>

{#if isLoggedIn}
  <button>Logout</button>
{:else}
  <button>Login</button>
{/if}

```

```

<script>
  export let cats;
</script>

<ul>
  {#each cats as cat}
    <li><a target="_blank" href="{cat.url}">
      {cat.name}
    </a></li>
  {/each}
</ul>

```


Comparando código

React

```
import React, { useState } from 'react';

const App = () => {
  const [count, setCount] = useState(0);

  return (
    <button onClick={() => setCount(count + 1)}>
      Clicked: {count}
    </button>
  );
}

export default App;
```

Svelte

```
<script>
  let count = 0;
</script>

<button on:click={() => count += 1}>
  Clicked: {count}
</button>
```

Vue

```
<template>
  <button v-on:click="count++">
    Clicked: {{count}}
  </button>
</template>

<script>
  export default {
    data() {
      return { count: 0 }
    }
  }
</script>
```



Talk is cheap. Show me the code.

— *Linus Torvalds* —

Svelte Store



Qual gerenciador de estado usar no React?



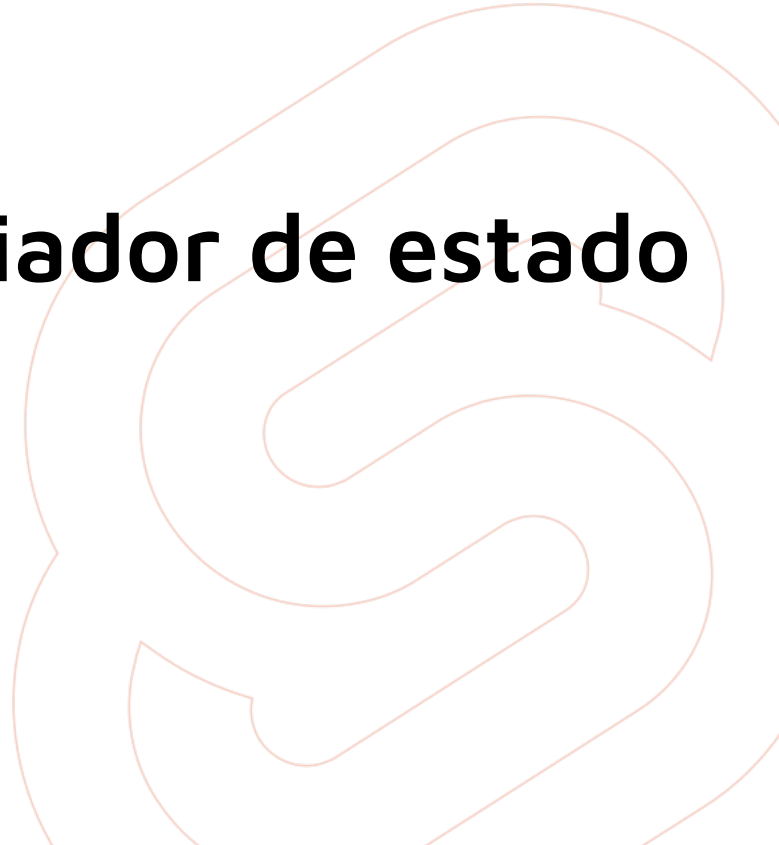
Effector



RECOIL



Svelte Store é o gerenciador de estado nativo do svelte.



Características do svelte store

- Faz parte do svelte e não é uma biblioteca extra
- Simplicidade
- Documentação clara

```
import { writable } from 'svelte/store';  
  
export const count = writable(0);
```

```
<script>  
  import { count } from './stores.js';  
  
  function decrement() {  
    count.update(n => n - 1);  
  }  
</script>  
  
<button on:click={decrement}>  
  -  
</button>
```

**Usando
svelte dentro
de outros
frameworks**





Já ouviram falar do CapivaraJS?



capivara.js

por:  GUMGA

<https://github.com/CapivaraJS/capivara.js>

Framework híbrido e suas diretivas são totalmente personalizáveis, tornando assim os componentes independentes da tecnologia utilizada.

**Será que podemos fazer isso com
svelte?**

SIM



Svelte Adapter

Utilitário para usar seu componente Svelte dentro do React e Vue.

- Github: <https://github.com/pngwn/svelte-adapter>

```
import React, { useState } from "react";
import toReact from "svelte-adapter/react";

import SvelteApp from "../App.svelte";

const baseStyle = {
  width: "50%"
};

const SvelteInReact = toReact(SvelteApp, baseStyle, "div");

const App = () => {
  const [count, setCount] = useState(10);

  const handleClick = () => setCount(prevCount => prevCount + 1);

  return (
    <div>
      <SvelteInReact
        number={count}
        onMagicalClick={handleClick}
        watchNumber={n => setCount(n)}
      />
      <button onClick={handleClick}>Increment - {count}</button>
    </div>
  );
};
```

```
<template>
  <div>
    <SvelteInVue
      :number="count"
      @magicalclick="count += 1"
      @watch:number="setCount"
    />
    <button @click="handleClick">Increment - {{ count }}</button>
  </div>
</template>

<script>
import SvelteApp from "../App.svelte";
import toVue from "svelte-adapter/vue";

export default {
  components: {
    SvelteInVue: toVue(SvelteApp, baseStyle, "div")
  },
  data() {
    return { count: 0 };
  },
  methods: {
    setCount(n) { this.count = n; }
  }
};
</script>
```

Sapper

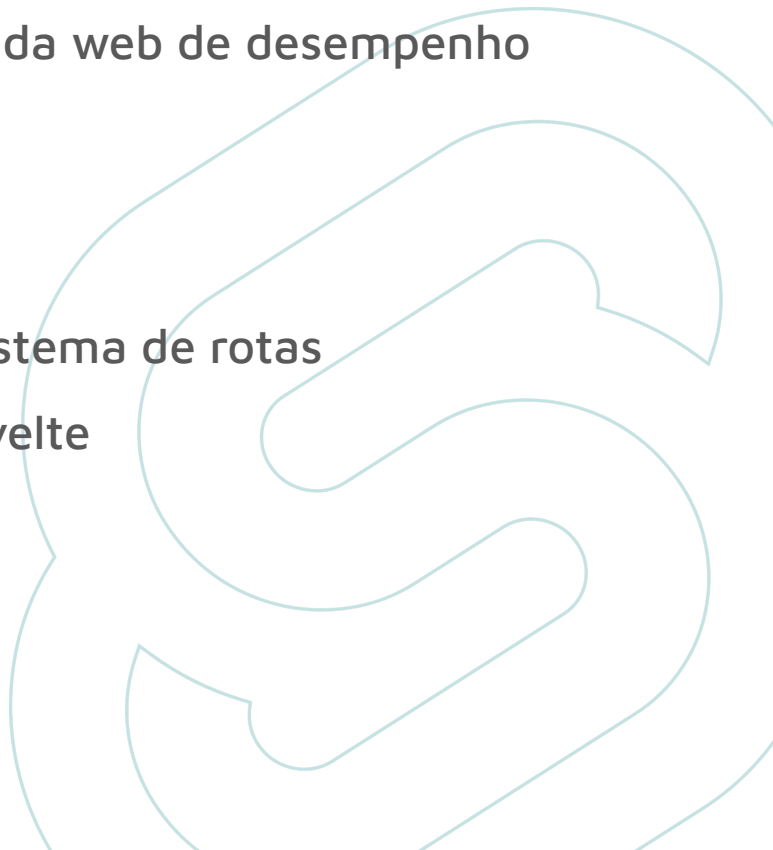
O SSR do Svelte



O que é esse sapper?

O Sapper é um framework para criar aplicativos da web de desempenho extremamente alto utilizando svelte.

- Inspirado no Next.js
- Estrutura de pastas definidas em cima do sistema de rotas
- Cada tela é basicamente um componente Svelte
- Não compromete o SEO
- Suporte ao modo offline
- Code Splitting

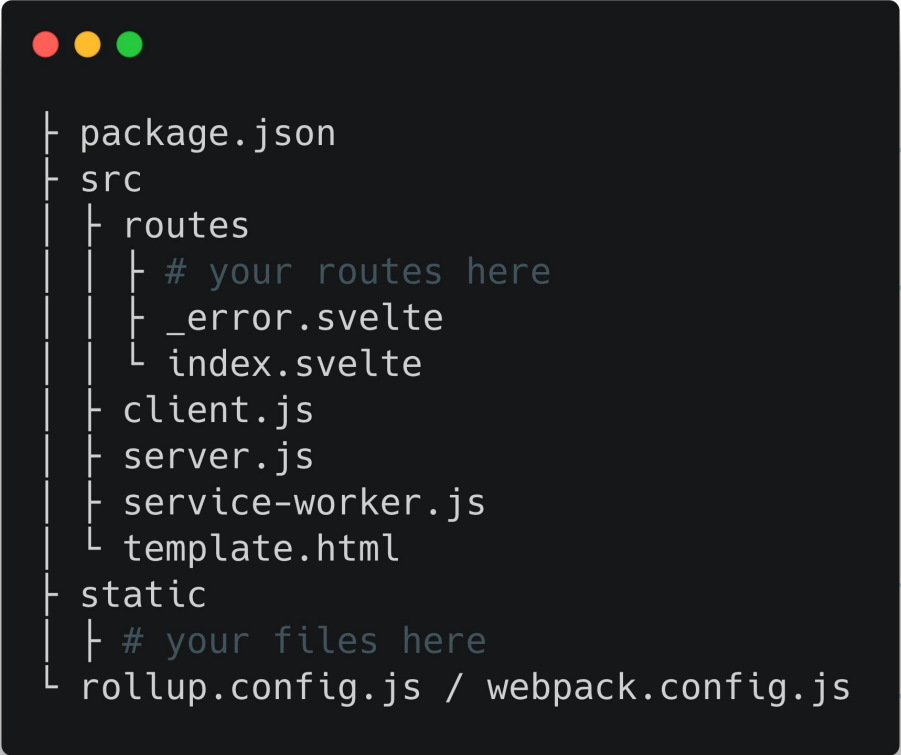


Como iniciar um projeto sapper?



```
npx degit "sveltejs/sapper-template#rollup" my-app  
# or: npx degit "sveltejs/sapper-template#webpack" my-app  
cd my-app  
npm install  
npm run dev
```

Estrutura de pastas de um projeto sapper

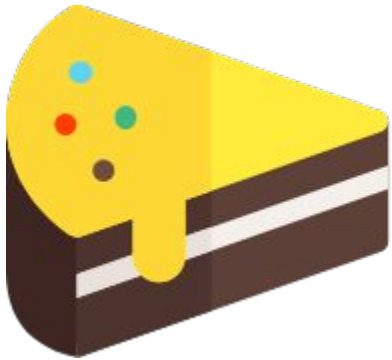


```
├─ package.json
├─ src
│  └─ routes
│     │ # your routes here
│     └─ _error.svelte
│        index.svelte
├─ client.js
├─ server.js
├─ service-worker.js
└─ template.html
├─ static
│  │ # your files here
└─ rollup.config.js / webpack.config.js
```

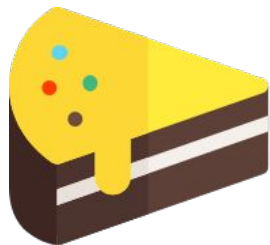
Revelt



E se compararmos métricas do svelte com outras tecnologias?



REVELT



REVELT

Analisa métricas de uma aplicação idêntica desenvolvida com diferentes tecnologias FrontEnd.

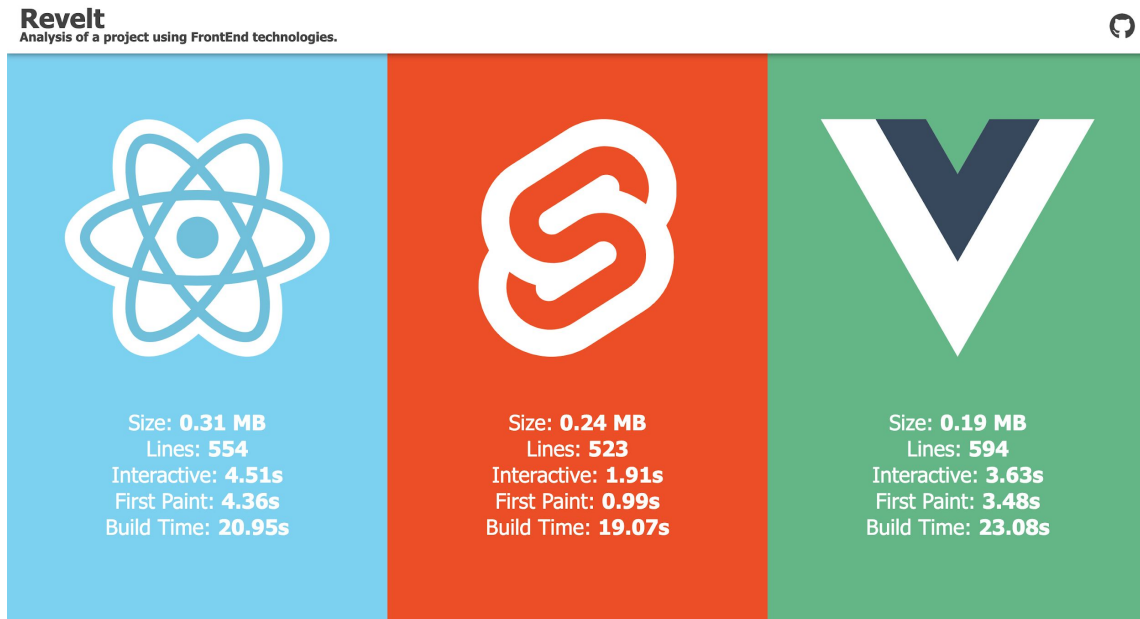
- Tecnologias React/Vue/Svelte (Em breve Angular e Preact)
- A aplicação é uma pokedéx.



get-folder-size

cloc

Como acessar o revelt?



Link: <https://revelt.netlify.app/>

Github: <https://github.com/andremlins/revelt>

Comunidade Svelte



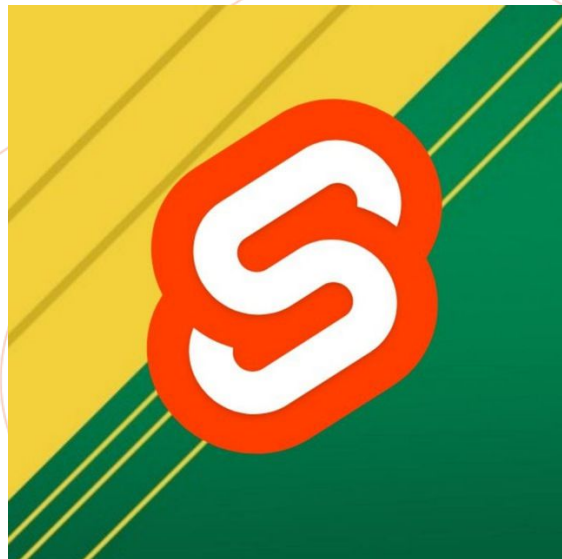
Comunidade svelte mundo

- <https://svelte-community.netlify.app/>
- Discord: <https://discord.com/invite/yy75DKs>
- Mais de 1000 membros no discord
- 15 grupos no mundo



Svelte Brasil

- Github: <https://github.com/svelte-brasil>
- Twitter: <https://twitter.com/sveltebrasil>
- Telegram:
 - 242 integrantes
 - Link: <https://t.me/sveltebrasil>
- Eventos
 - 1º meetup de Svelte do Brasil e da América Latina



Como criar bibliotecas para svelte?



Criando sua Biblioteca Svelte

<https://github.com/svelte-brasil/criando-sua-biblioteca-svelte>

Mas nem tudo são flores...



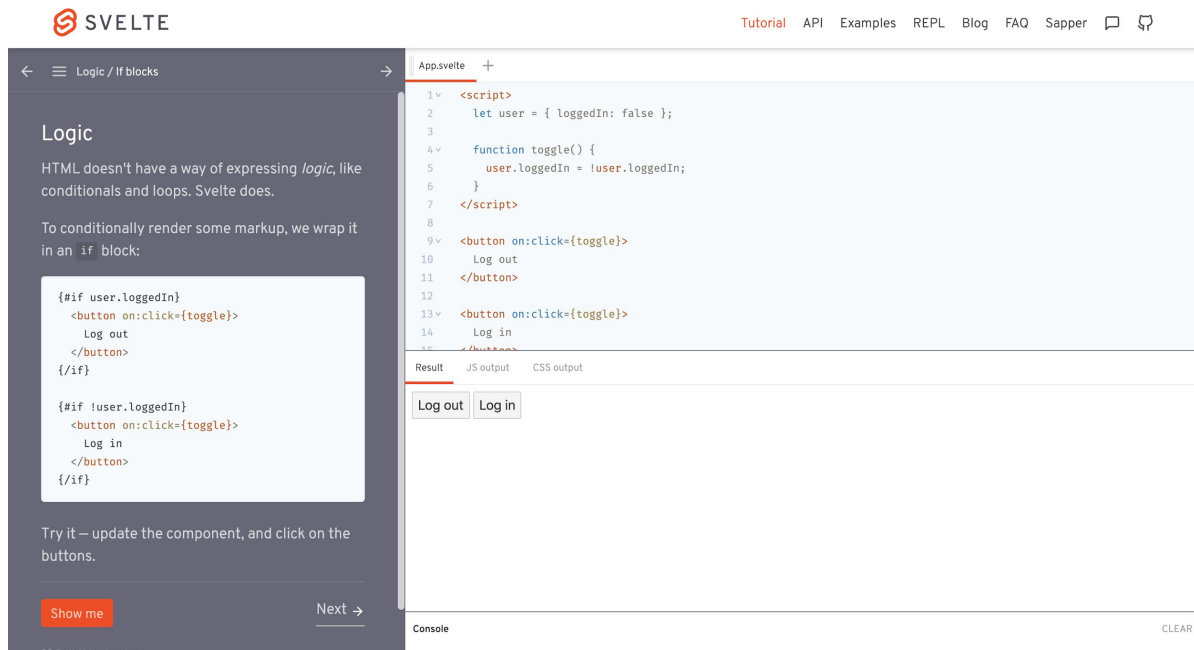
Pontos de melhoria e atenção

- Ainda existem poucas bibliotecas
- Comunidade é pequena e o seu time também
- Por ser um compilador é mais possível que exista bugs
- A melhora de performance que o svelte traz pode não ser interessante em vários casos.
- Gerar código javascript pode fazer o seu bundle ficar maior em grandes aplicações.

Quer aprender mais sobre Svelte?



Como aprender Svelte?



The screenshot displays the Svelte.dev website's tutorial interface. On the left, a sidebar titled 'Logic' explains that HTML lacks native logic and that Svelte uses `if` blocks for conditional rendering. It provides two code snippets: one for rendering a 'Log out' button when `user.loggedIn` is true, and another for rendering a 'Log in' button when it is false. Below the snippets, a 'Show me' button and a 'Next' link are visible. The main area on the right shows the Svelte component code for `App.svelte`, which includes a `toggle` function and two buttons: 'Log out' and 'Log in'. Below the code editor, tabs for 'Result', 'JS output', and 'CSS output' are shown, with 'Result' selected, displaying the rendered buttons. The top navigation bar includes links for 'Tutorial', 'API', 'Examples', 'REPL', 'Blog', 'FAQ', 'Sapper', and social media icons.

Logic

HTML doesn't have a way of expressing *logic*, like conditionals and loops. Svelte does.

To conditionally render some markup, we wrap it in an `if` block:

```
{#if user.loggedIn}
  <button on:click={toggle}>
    Log out
  </button>
{/if}

{#if !user.loggedIn}
  <button on:click={toggle}>
    Log in
  </button>
{/if}
```

Try it – update the component, and click on the buttons.

[Show me](#) [Next →](#)

```
1 <script>
2   let user = { loggedIn: false };
3
4   function toggle() {
5     user.loggedIn = !user.loggedIn;
6   }
7 </script>
8
9 <button on:click={toggle}>
10   Log out
11 </button>
12
13 <button on:click={toggle}>
14   Log in
15 </button>
```

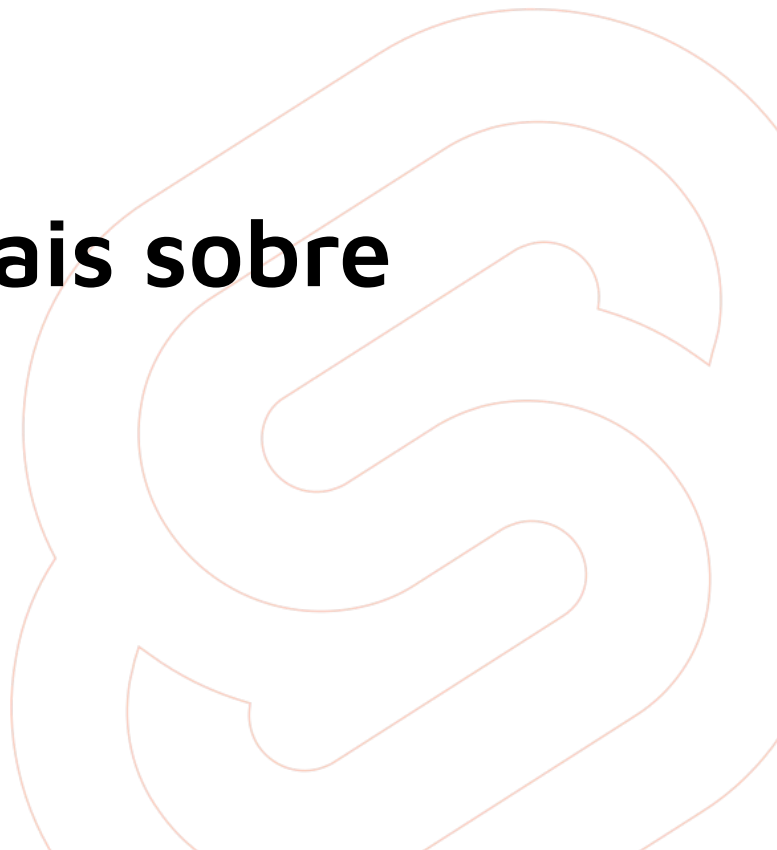
Result JS output CSS output

Log out Log in

Console CLEAR

<https://svelte.dev/tutorial>

**Quer aprender ainda mais sobre
svelte?**



ON-LINE

Svelte Society Meetup

01 Setembro

2020, Terça-Feira

19:30

HORÁRIO DE BRASÍLIA



B R A Z I L
SVELTE
S O C I E T Y

REGISTRE-SE AGORA:
[HTTP://TINY.CC/MEETUPS
VELTEBRASIL2](http://tiny.cc/meetupsveltebrasil2)

LIGHTNING TALKS:

Demys
Cota

**ESTILIZAÇÃO DINÂMICA
COM SVELTE DO JEITO
FÁCIL**

Henrique
Rodrigues

**CONFIGURAÇÃO DE
ARQUITETURA DE PROJETO
E GERAÇÃO INSTANTÂNEA
DE COMPONENTES NO
SVELTE**

**venha para
esse lado
da força...**





ANDRÉ LINS

- Engenheiro FrontEnd Sênior na PagSeguro
- Graduado em Ciência da Computação pela UFRPE
- Pós-Graduando em Engenharia de Software pela PUC Minas
- Viciado em OpenSource
- Desenvolvedor de algumas bibliotecas para Svelte, React, Vue, Node, ReasonML, Rust entre outras tecnologias.
- Tentando não ser evangelista Javascript

<https://twitter.com/andremlins>

<https://github.com/andremlins>