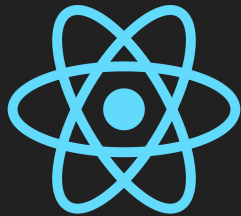
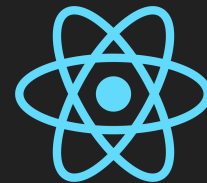


# JavaScript Isomórfico utilizando ReactJS



# <About />



Diego Marques

23 anos

10 anos de Desenvolvimento

1 ano e meio de React

GitHub

<https://github.com/dmarquesdev>

E-mail

[dmarquesdev@gmail.com](mailto:dmarquesdev@gmail.com)

Linkedin

<https://www.linkedin.com/in/diego-marques-17a92235/>

Repositório - TCC

<https://github.com/dmarquesdev/isomorphic-javascript>

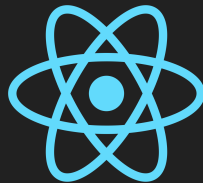
Repositório - Apresentação

<https://github.com/dmarquesdev/isomorphic-react>

# <JavaScript /> && <Isomorphism />

- JavaScript: Linguagem utilizada no ~~front-end~~, no ~~back-end~~, em qualquer-end para ~~validar formulários~~, para ~~criar aplicações Web~~, para criar qualquer coisa, onde surgem aproximadamente 3 frameworks novos por hora.  
(PS: enquanto isso era lido, pelo menos mais um framework foi criado).

<https://github.com/dmarquesdev/isomorphic-react>

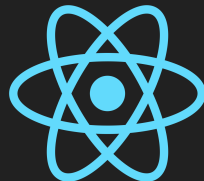


# <JavaScript /> && <Isomorphism />

- Isomorfismo: Palavra que deriva do grego onde a parte “Iso” significa “igual” e “morphos” significa “forma”.  
O isomorfismo então significa que.... ZZZZZZZ



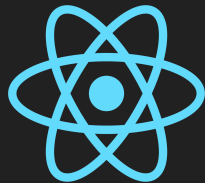
<https://github.com/dmarquesdev/isomorphic-react>



# <JavaScript show="history" />

- **Primórdios**: Manipulação de DOM e pequenos cálculos.
- **Web 2.0**: Validação de formulários
- **AJAX**: Atualização de conteúdo na página sem recarregar.
- **Chrome V8**: Possibilidade de executar JavaScript no servidor
- **SPAs**: Aplicações front-end completas utilizando JavaScript e AJAX

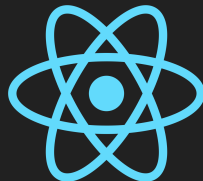
<https://github.com/dmarquesdev/isomorphic-react>



## <NodeJS />

- Implementação baseada no Chrome V8, da Google.
- Basicamente “tirou” o interpretador JavaScript do Chrome e deu a possibilidade de executar código da linguagem no servidor.
- Aplicações Back-end podem ser escritas utilizando JavaScript a partir deste momento.

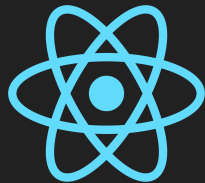
<https://github.com/dmarquesdev/isomorphic-react>



## <SinglePageApplication />

- Modelo de aplicação Front-end com a interação completamente client-side.
- O Back-end fornece a aplicação na primeira requisição e a comunicação acontece via AJAX (na maioria dos casos)
- Melhoria na usabilidade e maior fluidez ao usuário, retirando a necessidade de recarregar a aplicação durante a navegação

<https://github.com/dmarquesdev/isomorphic-react>

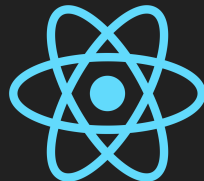


## <SPA problems="true" />

- Conteúdo da página é gerado dinamicamente pelo JavaScript durante a execução, o que dificulta a indexação pelas ferramentas de busca.
- Carga de processamento unilateral, podendo causar gargalos baseados no hardware do usuário.
- Enquanto o browser faz o download do “*app.js*” o usuário é apresentado como uma tela em branco/tela de loading.
- Toda a lógica de navegação e autorização ficam no Front-end, o que pode ser perigoso!!

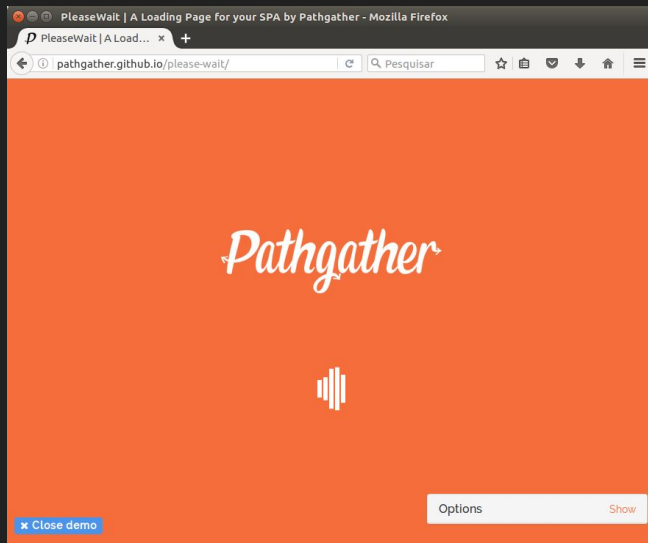
(First Rule of Secure Programming: NEVER TRUST USER INPUT!!!)

<https://github.com/dmarquesdev/isomorphic-react>

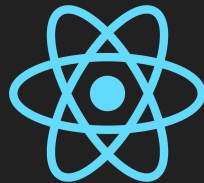




<SPA problems="true" />



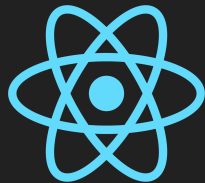
<https://github.com/dmarquesdev/isomorphic-react>



## <IsomorphicJS />

- Na primeira requisição HTTP do browser ao servidor, o servidor **executa** a aplicação “*app.js*” e retorna o HTML gerado, junto com o próprio “*app.js*” e suas dependências em sua resposta.

<https://github.com/dmarquesdev/isomorphic-react>



# <IsomorphicJS />

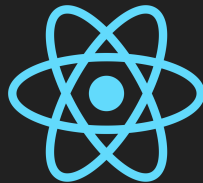
- = ~~Sem conteúdo para apresentar para os buscadores~~

A primeira requisição retorna o conteúdo do HTML já renderizado.

- = ~~Sem balanceamento de carga de processamento~~

O processamento é feito inicialmente no servidor, e o HTML retornado (quando utilizado o React) é otimizado de uma forma que o browser tem menos trabalho para fazer o que é necessário

<https://github.com/dmarquesdev/isomorphic-react>



# <IsomorphicJS />

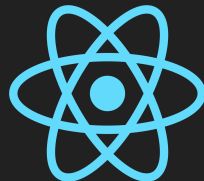
- = ~~Tela em branco/Loading enquanto carrega~~

A primeira requisição retorna o conteúdo do HTML já renderizado, sendo que apenas a interação não estará completa, logo o usuário verá algum conteúdo no momento em que ocorrer a requisição.

- = ~~Lógica de navegação completamente client-side~~

A lógica de navegação pode ser adaptada para ser filtrada no servidor.

<https://github.com/dmarquesdev/isomorphic-react>

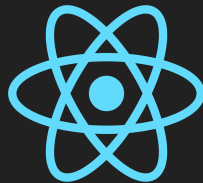


# <SSR /> vs <IsomorphicJS /> vs <UniversalJS />

- **Aplicação Isomórfica**

Transcende o JavaScript, e tem o significado de uma aplicação que pode ser rodada em diversos ambientes (servidor, desktop, browser, etc). Podemos ter uma aplicação escrita com React rodando tanto no browser quando em um servidor Rails, por exemplo. **(Tudo é possível, basta acreditar).**

<https://github.com/dmarquesdev/isomorphic-react>



# <SSR /> vs <IsomorphicJS /> vs <UniversalJS />

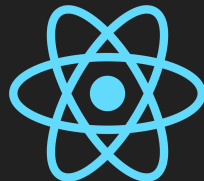
- JavaScript Isomórfico

Aplicação JavaScript capaz de rodar em ambientes que tem a capacidade de executar JavaScript (node, browser, etc.)

- JavaScript Universal

Aplicação escrita inteiramente em JavaScript que tem a capacidade de rodar em diversos ambientes

<https://github.com/dmarquesdev/isomorphic-react>



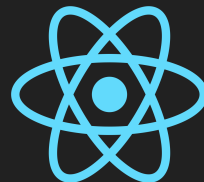
# <SSR /> vs <IsomorphicJS /> vs <UniversalJS />

- Server-side Rendering

Aplicação que é renderizada no servidor



<https://github.com/dmarquesdev/isomorphic-react>

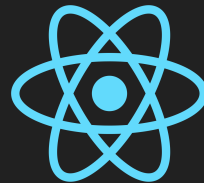


## <Conclusion />

Não existe um termo ideal que descreve exatamente o que está acontecendo no momento, então escolha o seu e seja feliz!!!

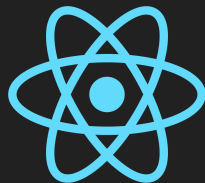


<https://github.com/dmarquesdev/isomorphic-react>





# <HandsON />



<https://github.com/dmarquesdev/isomorphic-react>

# <ThankYou />

## Dúvidas? Perguntas

[dmarquesdev@gmail.com](mailto:dmarquesdev@gmail.com)



<https://github.com/dmarquesdev/isomorphic-react>