

# Hotwire

by Jackson Pires



**Por que Hotwire?**

# Por que Hotwire?

- Hotwire é uma nova abordagem na construção de aplicações web por meio do envio de HTML sobre a conexão, com técnicas "diferente" das tradicionais adotados por outros frameworks de front-end a fim de conseguir um equilíbrio entre usar somente sintaxe ruby/rails e Javascript.

**O que veremos neste curso?**

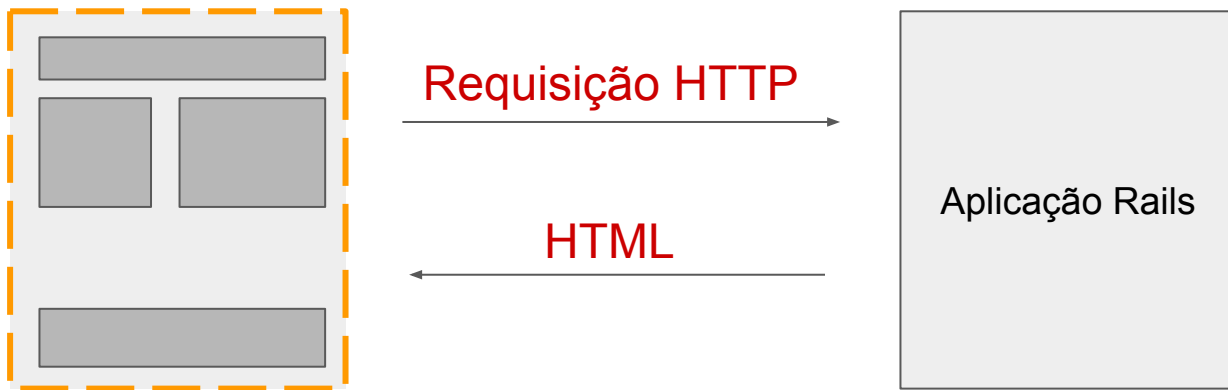
# O que veremos neste curso?

- Hotwire
  - Turbo Drive
  - Turbo Frame
  - Turbo Streams
  - Stimulus
  - ~~— Strada~~

# **Entendendo o que é e onde o Hotwire se encaixa**

# Entendendo o que é e onde o Hotwire se encaixa

- Como as coisas funcionam no Rails/Web.
  - 1) Método tradicional. A página HTML é totalmente renderizada no navegador
    - Neste método, toda a página é carregada a cada clique, mesmo que a necessidade seja apenas a atualização de parte da página



# Entendendo o que é e onde o Hotwire se encaixa

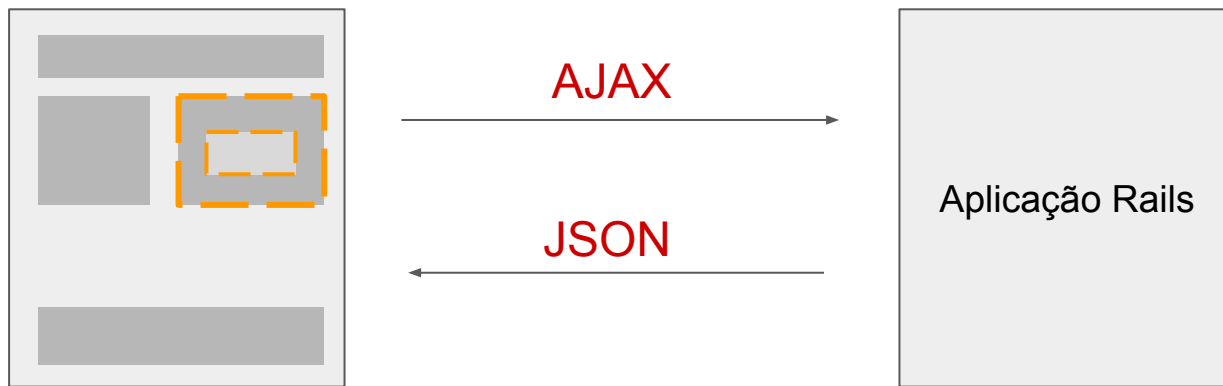
- Como as coisas funcionam no Rails/Web
  - 2) Método turbolinks. A página HTML é renderizada parcialmente no navegador
    - Neste método, o turbolinks se encarrega de a qualquer requisição, atualizar a head e body da página, evitando assim um reload completo.





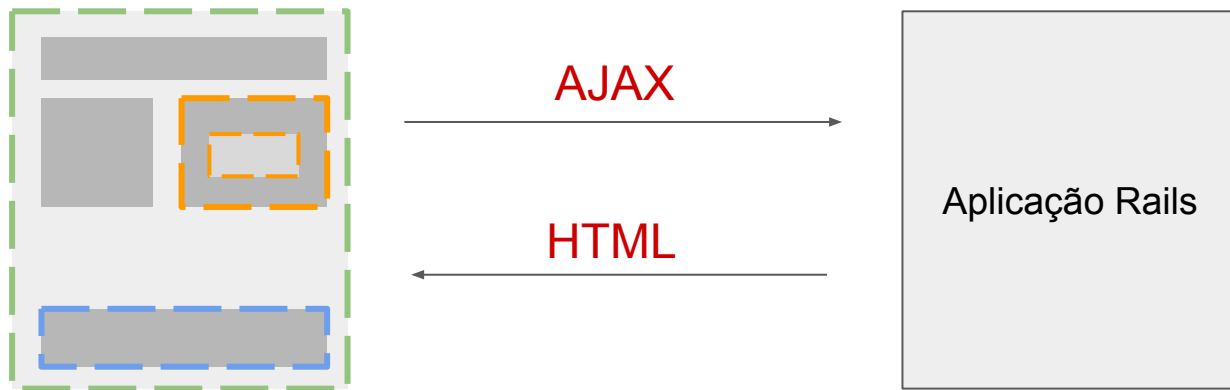
# Entendendo o que é e onde o Hotwire se encaixa

- Como as coisas funcionam na WEB
  - 3) Método componentes JS (React, Vue, etc). A página HTML tem elementos DOM que são carregados de forma independente.
    - Neste método, toda a página HTML é dividida em "componentes" que podem ser atualizada de forma independente e que possuem estado próprio.



# Entendendo o que é e onde o Hotwire se encaixa

- Como as coisas funcionam na WEB
  - Existem outras abordagens, mas é aqui onde o Hotwire nasce.
  - Imagine a possibilidade de ter os benefícios de renderizações parciais, sem a complexidade de manter estados entre front e back, além de poder usar templates Rails. Isso é o Hotwire!




# Entendendo cada parte do Hotwire

# Entendendo cada parte do Hotwire

- Hotwire
  - Turbo Drive
  - Turbo Frame
  - Turbo Streams
  - Stimulus

# Entendendo cada parte do Hotwire

- Hotwire
    - Turbo Drive
    - Turbo Frame
    - Turbo Streams
    - Stimulus **JS**
- 

# Entendendo cada parte do Hotwire

- Hotwire
  - **Turbo Drive:** Tem o foco em controlar as requisições e substituir head e body das páginas, para que não seja necessário o reload. Em outras palavras, reload de página nunca mais!

# Entendendo cada parte do Hotwire

- Hotwire
  - **Turbo Drive:** Tem o foco em controlar as requisições e substituir head e body das páginas, para que não seja necessário o reload. Em outras palavras, reload de página nunca mais!
  - **Turbo Frame:** Não precisar de reload na página já é muito bom, mas e se pudéssemos carregar e atualizar partes da página de forma independente e tudo isso sem precisar de JS. Isso é o Turbo Frame!

# Entendendo cada parte do Hotwire

- Hotwire
  - **Turbo Drive:** Tem o foco em controlar as requisições e substituir head e body das páginas, para que não seja necessário o reload. Em outras palavras, reload de página nunca mais!
  - **Turbo Frame:** Não precisar de reload na página já é muito bom, mas e se pudéssemos carregar e atualizar partes da página de forma independente e tudo isso sem precisar de JS. Isso é o Turbo Frame!
  - **Turbo Streams:** E se fosse possível atualizar várias partes da página ao mesmo tempo, sem usar JS. Isso é o Turbo Streams!



# Entendendo cada parte do Hotwire

- Hotwire
  - **Turbo Drive:** Tem o foco em controlar as requisições e substituir head e body das páginas, para que não seja necessário o reload. Em outras palavras, reload de página nunca mais!
  - **Turbo Frame:** Não precisar de reload na página já é muito bom, mas e se pudéssemos carregar e atualizar partes da página de forma independente e tudo isso sem precisar de JS. Isso é o Turbo Frame!
  - **Turbo Streams:** E se fosse possível atualizar várias partes da página ao mesmo tempo, sem usar JS. Isso é o Turbo Streams!
  - **Stimulus JS:** Apesar de termos todos os benefícios do Turbo, em alguns momentos precisaremos interagir com bibliotecas JS ou ainda criar interações específicas, sem requisições ao servidor, para dar vida à nossa página e assim nasce o Stimulus JS, para permitir que façamos qualquer tipo de interação necessária em nossa página, tudo isso usando os mesmos conceitos de controller e actions que já conhecemos do Rails.

# Documentação

# Documentação

- Site oficial
  - <https://hotwired.dev/>