

Introdução ao Ruby on Rails

by Jackson Pires



Entendendo as Rotas + Controllers

Rotas

- Documentação:
 - <https://guides.rubyonrails.org/routing.html>

Rotas

- Documentação:
 - <https://guides.rubyonrails.org/routing.html>
- Verificando as rotas da aplicação
 - `$> rails routes`
 - `http://localhost:3000/rails/info/routes`

Rotas

- Documentação:
 - <https://guides.rubyonrails.org/routing.html>
- Verificando as rotas da aplicação:
 - `$> rails routes`
 - <http://localhost:3000/rails/info/routes>
- Definindo novas rotas:
 - `config/routes.rb`
 - `get '/path', to: 'controller#action'`
 - Ex: **`get '/usuarios', to: 'users#index'`**
 - Ex: **`get '/inicio', to: 'users#index'`**

Rotas

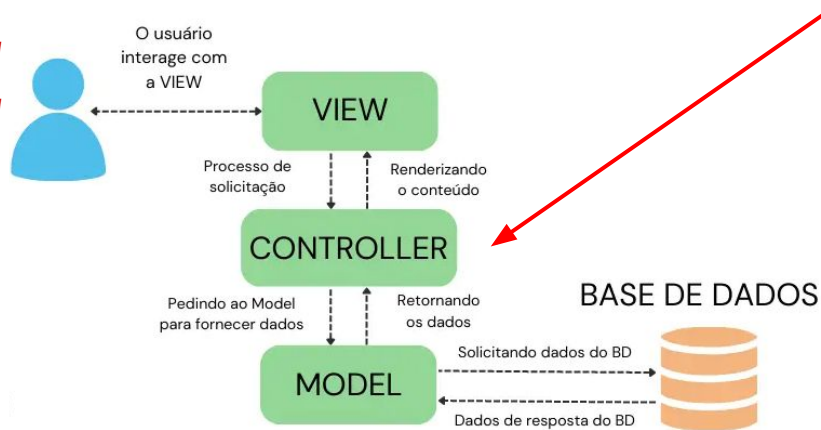
- Definindo a URL raiz:
 - `root "users#index"`

Resources, verbos HTTP e REST

Rotas

- Criando um controller
 - `app/controllers/home_controller.rb`

```
class HomeController < ApplicationController
  def index
    puts "CHEGUEI AQUI"
    puts "-----"
  end
end
```



Resources, verbos HTTP e REST

- Resources, verbos HTTP e CRUD:
 - <https://guides.rubyonrails.org/routing.html#resource-routing-the-rails-default>

Resources, verbos HTTP e REST

- Resources, verbos HTTP e CRUD:
 - <https://guides.rubyonrails.org/routing.html#resource-routing-the-rails-default>
- REST:
 - *REST (Representational State Transfer) se aplica à construção de Aplicações Web, como softwares e sites.*
 - *REST nada mais é que um conjunto de melhores práticas denominadas **constraints**.*
 - *Em outras palavras, as "regras REST" visam que os programadores utilizem essas regras para criar uma hierarquia bem organizada do código.*

Entendendo as Controllers e Views

Entendendo as Controllers e Views

- Controllers
 - Localizam-se em `app/controllers`
 - Sufixo `_controller.rb`
 - Toda **action** de um controller possui uma **view** correspondente
 - Formatos (`.json`, `.xml`, `.pdf`, etc)
 - `respond_to do |format|`

Entendendo as Controllers e Views

- Controllers
 - Localizam-se em `app/controllers`
 - Sufixo `_controller.rb`
 - Toda **action** de um controller possui uma **view** correspondente
 - Formatos (.json, .xml, .pdf, etc)
 - `respond_to do |format|`
- Views
 - **Partials** são "pedaços de código" que possuem o prefixo **underline** "_" no nome do arquivo
 - Respondem ao formato solicitado