



**Curso Completo**

**Aula 034**

**Autorização com Pundit**

**Parte 1/2**



**Pundit**



# **Autenticação vs Autorização**

# Pundit

<https://github.com/elabs/pundit>

# Instalação

```
gem 'pundit'
```

```
include Pundit (Application Controller)
```

```
rails g pundit:install (app/policies)
```

# Errors

<https://github.com/elabs/pundit#rescuing-a-denied-authorization-in-rails>



**TEORIA**



# Policies

<https://github.com/elabs/pundit#policies>

# Policies

- Pasta `app/policies`
- PORO
- As classes possuem o mesmo nome do `model` com o sufixo “Policy”

# Policies

- O primeiro argumento do construtor deve ser o usuário. O Pundit chamará o método **current\_user** para saber quem chamou o método. Na classe Pundit o usuário é o atributo **user**.

# Policies

- O segundo argumento é um tipo de objeto do **model** ao qual você quer checar a autorização. Na verdade não necessariamente precisa ser do **ActiveRecord**. Na classe Pundit o **model** é o atributo **record**.

# Policies

- A classe deve implementar um método que faça a verificação. Geralmente esse método é uma ação mapeada do controller.

# Generator

```
rails g pundit:policy <model>  
rails g pundit:policy Admin
```

# Pundit - Controller

```
authorize <instance model/AR>
```

# Policy

```
def new?  
  user.full_access?  
end
```



# pundit\_user

<https://github.com/elabs/pundit#customize-pundit-user>

```
def pundit_user
  current_admin
end
```



# Turbolinks

# Remover Turbolinks

```
gem 'turbolinks'  
  
//= require turbolinks  
  
"data-turbolinks-track" => true
```

# Pundit - View

```
if policy(@admins).new?  
  #  
end
```