

Curso de Ruby on Rails - aulas 41 a 50

Jackson Pires

<https://www.youtube.com/watch?v=ZHPondVB9RQ&list=PLe3LRfCs4go-mkvHRMSXEOG-HDbzesyaP>

Resumo do curso feito por Roberto Pinheiro

AULA 42 - TESTES COM RAILS - PARTE 1/2

Testes unitários

Teste de unidade é toda a aplicação de teste nas assinaturas de entrada e saída de um sistema. Consiste em validar dados válidos e inválidos via I/O (entrada/saída) sendo aplicado por desenvolvedores ou analistas de teste. Uma unidade é a menor parte testável de um programa de computador.

Testes de integração

Teste de integração é a fase do teste de software em que módulos são combinados e testados em grupo. Ela sucede o teste de unidade, em que os módulos são testados individualmente, e antecede o teste de sistema, em que o sistema completo (integrado) é testado num ambiente que simula o ambiente de produção.

Testes de aceitação

Teste de aceitação é uma fase do processo de teste em que um teste de caixa-preta é realizado num sistema antes de sua disponibilização. Tem por função verificar o sistema em relação aos seus requisitos originais, e às necessidades atuais do usuário.

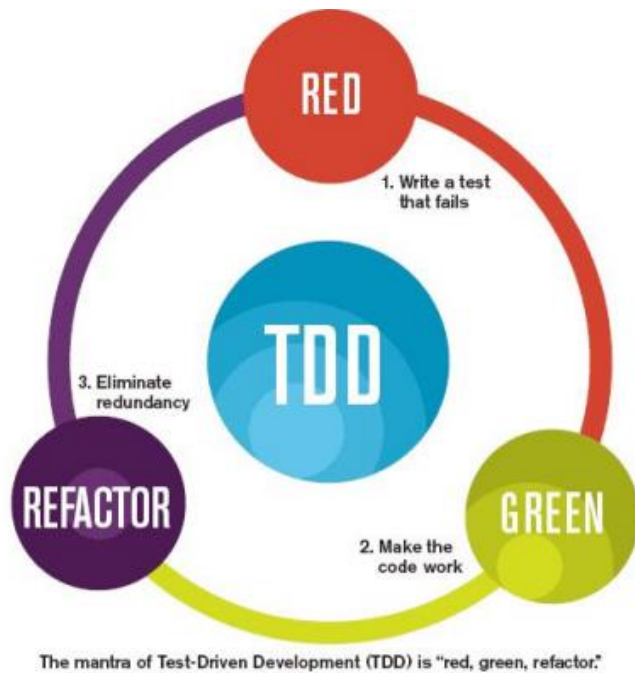
Testes unitários e de integração

- TestUnit <= Rails 3
- MiniTest >= Rails 4
- RSpec

Testes de aceitação

- Capybara

TDD (Test Driven Development)



Para não criar a pasta "teste" com MiniTest usar -T

```
rails new ror-jp-p4 -T --database=mysql
```

RSPEC

Instalação

```
group :development, :test do
  gem 'rspec-rails', '~> 3.0'
end
```

Gemfile

```
source 'https://rubygems.org'
git_source(:github) { |repo| "https://github.com/#{repo}.git" }

ruby '2.5.1'

# Bundle edge Rails instead: gem 'rails', github: 'rails/rails'
gem 'rails', '~> 5.2.3'
# Use mysql as the database for Active Record
```

```

gem 'mysql2', '>= 0.4.4', '< 0.6.0'
# Use Puma as the app server
gem 'puma', '~> 3.11'
# Use SCSS for stylesheets
gem 'sass-rails', '~> 5.0'
# Use Uglifier as compressor for JavaScript assets
gem 'uglifier', '>= 1.3.0'
# See https://github.com/rails/execjs#readme for more supported runtimes
gem 'duktape'
# Use CoffeeScript for .coffee assets and views
gem 'coffee-rails', '~> 4.2'
# Turbolinks makes navigating your web application faster. Read more:
https://github.com/turbolinks/turbolinks
gem 'turbolinks', '~> 5'
# Build JSON APIs with ease. Read more: https://github.com/rails/jbuilder
gem 'jbuilder', '~> 2.5'
# Use Redis adapter to run Action Cable in production
# gem 'redis', '~> 4.0'
# Use ActiveSupport has_secure_password
# gem 'bcrypt', '~> 3.1.7'

# Use ActiveSupport variant
# gem 'mini_magick', '~> 4.8'

# Use Capistrano for deployment
# gem 'capistrano-rails', group: :development

# Reduces boot times through caching; required in config/boot.rb
gem 'bootsnap', '>= 1.1.0', require: false

group :development, :test do
  # Call 'byebug' anywhere in the code to stop execution and get a debugger console
  gem 'byebug', platforms: [:mri, :mingw, :x64_mingw]
  gem 'rspec-rails', '~> 3.0'
end

group :development do
  # Access an interactive console on exception pages or by calling 'console' anywhere in the code.
  gem 'web-console', '>= 3.3.0'
end

# Windows does not include zoneinfo files, so bundle the tzinfo-data gem
gem 'tzinfo-data', platforms: [:mingw, :mswin, :x64_mingw, :jruby]

```

bundle install

rails generate rspec:install

```
C:\Sites\ror-jp-p4>rails generate rspec:install
create  .rspec
create  spec
create  spec/spec_helper.rb
create  spec/rails_helper.rb
```

.rspec

```
--color
--require spec_helper
--format documentation
```

Configurações adicionais

config/application.rb :

```
config.generators do |g|
  g.test_framework :rspec
end
```

config/application.rb

```
require_relative 'boot'

require "rails"
# Pick the frameworks you want:
require "active_model/railtie"
require "active_job/railtie"
require "active_record/railtie"
require "active_storage/engine"
require "action_controller/railtie"
require "action_mailer/railtie"
require "action_view/railtie"
require "action_cable/engine"
require "sprockets/railtie"
# require "rails/test_unit/railtie"

# Require the gems listed in Gemfile, including any gems
# you've limited to :test, :development, or :production.
Bundler.require(*Rails.groups)

module RorJpP4
```

```

class Application < Rails::Application
  # Initialize configuration defaults for originally generated Rails version.
  config.load_defaults 5.2

  # Settings in config/environments/* take precedence over those specified here.
  # Application configuration can go into files in config/initializers
  # -- all .rb files in that directory are automatically loaded after loading
  # the framework and any gems in your application.

  # Don't generate system test files.
  config.generators.system_tests = nil

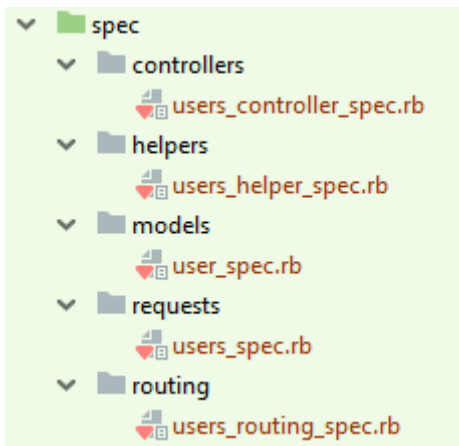
  config.generators do |g|
    g.test_framework :rspec
  end

end
end

```

Gerando teste

`rails g scaffold User name:string email:string`



`rake db:create db:migrate`

```

C:\Sites\ror-jp-p4>rake db:create db:migrate
Created database 'ror-jp-p4_development'
Created database 'ror-jp-p4_test'
== 20191027212048 CreateUsers: migrating =====
-- create_table(:users)
   -> 0.3524s
== 20191027212048 CreateUsers: migrated (0.3532s) =====

```

rake spec

```
C:\Sites\ror-jp-p4>rake spec
C:/Ruby25-x64/bin/ruby.exe -I'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/lib';'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/exe/rspec' --pattern 'spec/**{,/*/*}/*_spec.rb'

UsersController
  GET #index
    returns a success response (PENDING: Add a hash of attributes valid for your model)
  GET #show
    returns a success response (PENDING: Add a hash of attributes valid for your model)
  GET #new
    returns a success response
  GET #edit
    returns a success response (PENDING: Add a hash of attributes valid for your model)
  POST #create
    with valid params
      creates a new User (PENDING: Add a hash of attributes valid for your model)
      redirects to the created user (PENDING: Add a hash of attributes valid for your model)
    with invalid params
      returns a success response (i.e. to display the 'new' template) (PENDING: Add a hash of attributes invalid for your model)
  PUT #update
    with valid params
      updates the requested user (PENDING: Add a hash of attributes valid for your model)
      redirects to the user (PENDING: Add a hash of attributes valid for your model)
    with invalid params
      returns a success response (i.e. to display the 'edit' template) (PENDING: Add a hash of attributes valid for your model)
  DELETE #destroy
    destroys the requested user (PENDING: Add a hash of attributes valid for your model)
    redirects to the users list (PENDING: Add a hash of attributes valid for your model)

UsersHelper
  add some examples to (or delete) C:/Sites/ror-jp-p4/spec/helpers/users\_helper\_spec.rb (PENDING: Not yet implemented)

User
  add some examples to (or delete) C:/Sites/ror-jp-p4/spec/models/user\_spec.rb (PENDING: Not yet implemented)

Users
  GET /users
    works! (now write some real specs)

Users
  GET /users
    works! (now write some real specs)

UsersController
  routing
    routes to #index
    routes to #new
    routes to #show
    routes to #edit
    routes to #create
    routes to #update via PUT
    routes to #update via PATCH
    routes to #destroy

users/edit
  renders the edit user form

users/index
  renders a list of users

users/new
```

```

    renders new user form

users/show
    renders attributes in <p>

Pending: (Failures listed here are expected and do not affect your suite's status)

  1) UsersController GET #index returns a success response
     # Add a hash of attributes valid for your model
     # ./spec/controllers/users_controller_spec.rb:45

  2) UsersController GET #show returns a success response
     # Add a hash of attributes valid for your model
     # ./spec/controllers/users_controller_spec.rb:53

  3) UsersController GET #edit returns a success response
     # Add a hash of attributes valid for your model
     # ./spec/controllers/users_controller_spec.rb:68

  4) UsersController POST #create with valid params creates a new User
     # Add a hash of attributes valid for your model
     # ./spec/controllers/users_controller_spec.rb:77

  5) UsersController POST #create with valid params redirects to the created user
     # Add a hash of attributes valid for your model
     # ./spec/controllers/users_controller_spec.rb:83

  6) UsersController POST #create with invalid params returns a success response (i.e. to display the 'new' template)
     # Add a hash of attributes invalid for your model
     # ./spec/controllers/users_controller_spec.rb:90

  7) UsersController PUT #update with valid params updates the requested user
     # Add a hash of attributes valid for your model
     # ./spec/controllers/users_controller_spec.rb:103

  8) UsersController PUT #update with valid params redirects to the user
     # Add a hash of attributes valid for your model
     # ./spec/controllers/users_controller_spec.rb:110

  9) UsersController PUT #update with invalid params returns a success response (i.e. to display the 'edit' template)
     # Add a hash of attributes invalid for your model
     # ./spec/controllers/users_controller_spec.rb:118

```

Fazendo apenas teste dos models

Configurações adicionais:

```

config.generators do |g|
  g.test_framework :rspec,
    :view_specs => false,
    :controller_specs => false,
    :helper_specs => false,
    :routing_specs => false,
    :request_specs => false
end

```

config/application.rb

```
require_relative 'boot'

require "rails"
# Pick the frameworks you want:
require "active_model/railtie"
require "active_job/railtie"
require "active_record/railtie"
require "active_storage/engine"
require "action_controller/railtie"
require "action_mailer/railtie"
require "action_view/railtie"
require "action_cable/engine"
require "sprockets/railtie"
# require "rails/test_unit/railtie"

# Require the gems listed in Gemfile, including any gems
# you've limited to :test, :development, or :production.
Bundler.require(*Rails.groups)

module RorJpP4
  class Application < Rails::Application
    # Initialize configuration defaults for originally generated Rails version.
    config.load_defaults 5.2

    # Settings in config/environments/* take precedence over those specified here.
    # Application configuration can go into files in config/initializers
    # -- all .rb files in that directory are automatically loaded after loading
    # the framework and any gems in your application.

    # Don't generate system test files.
    config.generators.system_tests = nil

    #RSpec configuration
    config.generators do |g|
      g.test_framework :rspec, :view_specs => false,
        :controller_specs => false,
        :helper_specs => false,
        :routing_specs => false,
        :request_specs => false
    end

  end
end
```


- Dentro da pasta spec apague todas as subpastas.

rails destroy scaffold User

rails g scaffold User name:string email:string

rake db:drop db:create db:migrate

rake spec

```
C:\Sites\ror-jp-p4>rake spec
C:/Ruby25-x64/bin/ruby.exe -I'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/lib';'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-support-3.9.0/lib' 'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/exe/rspec' --pattern 'spec/**{,/*/**}/*_spec.rb'

User
  add some examples to (or delete) C:/Sites/ror-jp-p4/spec/models/user_spec.rb (PENDING: Not yet implemented)

Pending: (Failures listed here are expected and do not affect your suite's status)

  1) User add some examples to (or delete) C:/Sites/ror-jp-p4/spec/models/user_spec.rb
     # Not yet implemented
     # ./spec/models/user_spec.rb:4

Finished in 0.15243 seconds (files took 30.85 seconds to load)
1 example, 0 failures, 1 pending
```

Validações

app/models/user.rb

```
class User < ApplicationRecord
  validates :name, presence: true
  validates :email, presence: true
end
```

Estrutura básica do Rspec

- context
- it

- Context tem a única finalidade de organizar seus testes.
- It especifica a ação a ser feita/testada

Exemplo:

```
context "Validade de um Usuário" do
  it "Usuário válido?" do
    ... teste ...
  end
end
```

expectations/matchers

spec/models/user_spec.rb

```
require 'rails_helper'

RSpec.describe User, type: :model do
  context "Validade de um Usuário" do
    it "Usuário válido?" do
      user = User.new
      user.name = "Jackson"
      user.email = "jackson@gmail.com"
      expect(user.valid?).to be_truthy
    end
  end
end
```

```
C:\Sites\nor-jp-p4>rake spec
C:/Ruby25-x64/bin/ruby.exe -I'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/lib';'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-support-3.9.0/lib' 'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/exe/rspec' --pattern 'spec/**{,/*/**}/*_spec.rb'

User
  Validade de um Usuário
    Usuário válido?

Finished in 0.0466 seconds (files took 38.04 seconds to load)
1 example, 0 failures
```

spec/models/user_spec.rb

```
require 'rails_helper'

RSpec.describe User, type: :model do
  context "Validade de um Usuário" do
    it "Usuário válido?" do
      user = User.new
      user.name = "Jackson"
      user.email = "jackson@gmail.com"
      expect(user.valid?).to be_truthy
    end

    it "Usuário inválido?" do
      user = User.new
      user.name = "Jackson"
      user.email = nil
      expect(user.valid?).to be_falsey
    end
  end
end
```

```
C:\Sites\ror-jp-p4>rake spec
C:/Ruby25-x64/bin/ruby.exe -I'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/lib'; 'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-support-3.9.0/lib' 'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/exe/rspec' --pattern 'spec/**{,/**/*.rb}'

User
  Validade de um Usuário
    Usuário válido?
    Usuário inválido?

Finished in 0.28428 seconds (files took 29.28 seconds to load)
2 examples, 0 failures
```

spec/models/user_spec.rb

```
require 'rails_helper'

RSpec.describe User, type: :model do
  context "Validade de um Usuário" do
    it "Usuário válido?" do
      user = User.new
      user.name = "Jackson"
      user.email = "jackson@gmail.com"
      expect(user.valid?).to be_truthy
    end

    it "Usuário inválido?" do
      user = User.new
      user.name = "Jackson"
      user.email = nil
      expect(user.valid?).to be_falsey
    end
  end

  context "Títulos válidos" do
    it "Título completo" do
      user = User.new
      user.name = "Jackson"
      user.email = "jackson@gmail.com"
      expect(user.titulo_completo).to eq("Sr. Jackson - Email: jackson@gmail.com")
    end
  end
end
```

```
C:\Sites\ror-jp-p4>rake spec
C:/Ruby25-x64/bin/ruby.exe -I'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/lib'; 'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-support-3.9.0/lib' 'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/exe/rspec' --pattern 'spec/**/*.rb'

User
  Validade de um Usuário
    Usuário válido?
    Usuário inválido?
  Títulos válidos
    Título completo

Finished in 0.09092 seconds (files took 29.93 seconds to load)
3 examples, 0 failures
```

Links úteis

<https://www.relishapp.com>

<http://betterspecs.org/br/>

AULA 43 - TESTES COM RAILS - PARTE 2/2

Gemfile

```
source 'https://rubygems.org'
git_source(:github) { |repo| "https://github.com/#{repo}.git" }

ruby '2.5.1'

# Bundle edge Rails instead: gem 'rails', github: 'rails/rails'
gem 'rails', '~> 5.2.3'
# Use mysql as the database for Active Record
gem 'mysql2', '>= 0.4.4', '< 0.6.0'
# Use Puma as the app server
gem 'puma', '~> 3.11'
# Use SCSS for stylesheets
gem 'sass-rails', '~> 5.0'
# Use Uglifier as compressor for JavaScript assets
gem 'uglifier', '>= 1.3.0'
# See https://github.com/rails/execjs#readme for more supported runtimes
gem 'duktape'
# Use CoffeeScript for .coffee assets and views
gem 'coffee-rails', '~> 4.2'
# Turbolinks makes navigating your web application faster. Read more:
https://github.com/turbolinks/turbolinks
gem 'turbolinks', '~> 5'
# Build JSON APIs with ease. Read more: https://github.com/rails/jbuilder
gem 'jbuilder', '~> 2.5'
# Use Redis adapter to run Action Cable in production
gem 'redis', '~> 4.0'
# Use ActiveRecord has_secure_password
gem 'bcrypt', '~> 3.1.7'

# Use ActiveSupport variant
gem 'mini_magick', '~> 4.8'

# Use Capistrano for deployment
gem 'capistrano-rails', group: :development

# Reduces boot times through caching; required in config/boot.rb
gem 'bootsnap', '>= 1.1.0', require: false

group :development, :test do
  # Call 'byebug' anywhere in the code to stop execution and get a debugger console
  gem 'byebug', platforms: [:mri, :mingw, :x64_mingw]
  gem 'rspec-rails', '~> 3.0'
  gem 'capybara'
end
```

```
group :development do
  # Access an interactive console on exception pages or by calling 'console' anywhere in the code.
  gem 'web-console', '>= 3.3.0'
end

# Windows does not include zoneinfo files, so bundle the tzinfo-data gem
gem 'tzinfo-data', platforms: [:mingw, :mswin, :x64_mingw, :jruby]
```

Configuração Adicional

rails_helper.rb

config.include Capybara::DSL

spec/rails_helper.rb

```
# This file is copied to spec/ when you run 'rails generate rspec:install'
require 'spec_helper'
ENV['RAILS_ENV'] ||= 'test'

require File.expand_path('../config/environment', __dir__)

# Prevent database truncation if the environment is production
abort("The Rails environment is running in production mode!") if Rails.env.production?
require 'rspec/rails'
# Add additional requires below this line. Rails is not loaded until this point!

# Requires supporting ruby files with custom matchers and macros, etc, in
# spec/support/ and its subdirectories. Files matching `spec/**/*_spec.rb` are
# run as spec files by default. This means that files in spec/support that end
# in _spec.rb will both be required and run as specs, causing the specs to be
# run twice. It is recommended that you do not name files matching this glob to
# end with _spec.rb. You can configure this pattern with the --pattern
# option on the command line or in ~/.rspec, .rspec or .rspec-local.
#
# The following line is provided for convenience purposes. It has the downside
# of increasing the boot-up time by auto-requiring all files in the support
# directory. Alternatively, in the individual `*_spec.rb` files, manually
# require only the support files necessary.
#
# Dir[Rails.root.join('spec', 'support', '**', '*.rb')].each { |f| require f }

# Checks for pending migrations and applies them before tests are run.
# If you are not using ActiveRecord, you can remove these lines.
```

```

begin
  ActiveRecord::Migration.maintain_test_schema!
rescue ActiveRecord::PendingMigrationError => e
  puts e.to_s.strip
  exit 1
end
RSpec.configure do |config|
  # Remove this line if you're not using ActiveRecord or ActiveRecord fixtures
  config.fixture_path = "#{::Rails.root}/spec/fixtures"

  # If you're not using ActiveRecord, or you'd prefer not to run each of your
  # examples within a transaction, remove the following line or assign false
  # instead of true.
  config.use_transactional_fixtures = true

  # RSpec Rails can automatically mix in different behaviours to your tests
  # based on their file location, for example enabling you to call `get` and
  # `post` in specs under `spec/controllers`.
  #
  # You can disable this behaviour by removing the line below, and instead
  # explicitly tag your specs with their type, e.g.:
  #
  #   RSpec.describe UsersController, :type => :controller do
  #     # ...
  #   end
  #
  # The different available types are documented in the features, such as in
  # https://relishapp.com/rspec/rspec-rails/docs
  config.infer_spec_type_from_file_location!

  # Filter lines from Rails gems in backtraces.
  config.filter_rails_from_backtrace!
  # arbitrary gems may also be filtered via:
  # config.filter_gems_from_backtrace("gem name")

  # Capybara
  config.include Capybara::DSL
end

```

Configuração Adicional

environments/test.rb

config.allow_concurrency = false

config/environments/test.rb

```
Rails.application.configure do
  # Settings specified here will take precedence over those in config/application.rb.

  # The test environment is used exclusively to run your application's
  # test suite. You never need to work with it otherwise. Remember that
  # your test database is "scratch space" for the test suite and is wiped
  # and recreated between test runs. Don't rely on the data there!
  config.cache_classes = true

  # Do not eager load code on boot. This avoids loading your whole application
  # just for the purpose of running a single test. If you are using a tool that
  # preloads Rails for running tests, you may have to set it to true.
  config.eager_load = false

  # Configure public file server for tests with Cache-Control for performance.
  config.public_file_server.enabled = true
  config.public_file_server.headers = {
    'Cache-Control' => "public, max-age=#{1.hour.to_i}"
  }

  # Show full error reports and disable caching.
  config.consider_all_requests_local = true
  config.action_controller.perform_caching = false

  # Raise exceptions instead of rendering exception templates.
  config.action_dispatch.show_exceptions = false

  # Disable request forgery protection in test environment.
  config.action_controller.allow_forgery_protection = false

  # Store uploaded files on the local file system in a temporary directory
  config.active_storage.service = :test

  config.action_mailer.perform_caching = false

  # Tell Action Mailer not to deliver emails to the real world.
  # The :test delivery method accumulates sent emails in the
  # ActionMailer::Base.deliveries array.
  config.action_mailer.delivery_method = :test

  # Print deprecation notices to the stderr.
  config.active_support.deprecation = :stderr

  # Raises error for missing translations
  # config.action_view.raise_on_missing_translations = true
  config.allow_concurrency = false
end
```


Usando!

```
require 'capybara/rspec'
```

spec/models/user_spec.rb

```
require 'rails_helper'

RSpec.describe User, type: :model do

  context "Testando a View User" do
    it "Cadastrar um usuário" do
      visit '/users/new'
      within("#form_user") do
        fill_in 'Name', with: 'Jackson'
        fill_in 'Email', with: 'jackson@gmail.com'
      end
      click_button 'Create User'
      expect(page).to have_content 'User was successfully created.'
    end
  end

  context "Validade de um Usuário" do
    it "Usuário válido?" do
      user = User.new
      user.name = "Jackson"
      user.email = "jackson@gmail.com"
      expect(user.valid?).to be_truthy
    end

    it "Usuário inválido?" do
      user = User.new
      user.name = "Jackson"
      user.email = nil
      expect(user.valid?).to be_falsey
    end
  end

  context "Títulos válidos" do
    it "Título completo" do
      user = User.new
      user.name = "Jackson"
      user.email = "jackson@gmail.com"
      expect(user.titulo_completo).to eq("Sr. Jackson - Email: jackson@gmail.com")
    end
  end
end
```

app/views/users/_form.html.erb

```
<%= form_with(model: user, id: "form_user", local: true) do |form| %>
  <% if user.errors.any? %>
    <div id="error_explanation">
      <h2><%= pluralize(user.errors.count, "error") %> prohibited this user from being saved:</h2>

      <ul>
        <% user.errors.full_messages.each do |message| %>
          <li><%= message %></li>
        <% end %>
      </ul>
    </div>
  <% end %>

  <div class="field">
    <%= form.label :name %>
    <%= form.text_field :name %>
  </div>

  <div class="field">
    <%= form.label :email %>
    <%= form.text_field :email %>
  </div>

  <div class="actions">
    <%= form.submit %>
  </div>
<% end %>
```

rake spec

```
C:\Sites\ror-jp-p4>rake spec
C:/Ruby25-x64/bin/ruby.exe -I'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/lib';'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-support-3.9.0/lib' 'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/exe/rspec' --pattern 'spec/**/*.rb'

User
  Testando a View User
    Cadastrar um usuário
  Validade de um Usuário
    Usuário válido?
    Usuário inválido?
  Títulos válidos
    Título completo

Finished in 1.61 seconds (files took 35.29 seconds to load)
4 examples, 0 failures
```

spec/models/user_spec.rb

```
require 'rails_helper'

RSpec.describe User, type: :model do

  context "Testando a View User" do
    it "Cadastrar um usuário" do
      users = User.all
      size_old = users.size
      visit '/users/new'
      within("#form_user") do
        fill_in 'Name', with: 'Jackson'
        fill_in 'Email', with: 'jackson@gmail.com'
      end
      click_button 'Create User'
      expect(page).to have_content 'User was successfully created.'
      users = User.all
      expect(users.size).to eq(size_old + 1)
    end
  end

  context "Validade de um Usuário" do
    it "Usuário válido?" do
      user = User.new
      user.name = "Jackson"
      user.email = "jackson@gmail.com"
      expect(user.valid?).to be_truthy
    end

    it "Usuário inválido?" do
      user = User.new
      user.name = "Jackson"
      user.email = nil
      expect(user.valid?).to be_falsey
    end
  end

  context "Títulos válidos" do
    it "Título completo" do
      user = User.new
      user.name = "Jackson"
      user.email = "jackson@gmail.com"
      expect(user.titulo_completo).to eq("Sr. Jackson - Email: jackson@gmail.com")
    end
  end
end
```

```
C:\Sites\ror-jp-p4>rake spec
C:/Ruby25-x64/bin/ruby.exe -I'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/lib';'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-support-3.9.0/lib' 'C:/Ruby25-x64/lib/ruby/gems/2.5.0/gems/rspec-core-3.9.0/exe/rspec' --pattern 'spec/**/*.rb'

User
  Testando a View User
    Cadastrar um usuário
  Validade de um Usuário
    Usuário válido?
    Usuário inválido?
  Títulos válidos
    Título completo

Finished in 1.35 seconds (files took 33.62 seconds to load)
4 examples, 0 failures
```

AULA 44 - ASSET PIPELINE 1/2

O que é?

Asset Pipeline fornece uma estrutura para concatenar e minificar ou comprimir os assets JavaScript e CSS. Ele também adiciona a capacidade de escrever esses assets em outros idiomas e pré-processadores como CoffeeScript, Sass e ERB.

Organização

- app/assets
- lib/assets
- vendor/assets.

Usando

```
<%= stylesheet_link_tag "application", media: "all" %>
<%= javascript_include_tag "application" %>
```

app/views/layouts/application.html.erb

```
<!DOCTYPE html>
<html>
  <head>
    <title>RorJpP4</title>
    <%= csrf_meta_tags %>
    <%= csp_meta_tag %>

    <%= stylesheet_link_tag 'application', media: 'all', 'data-turbolinks-track': 'reload' %>
    <%= javascript_include_tag 'application', 'data-turbolinks-track': 'reload' %>
  </head>

  <body>
    <%= yield %>
  </body>
</html>
```

app/asset/javascripts/application.js

```
// This is a manifest file that'll be compiled into application.js, which will include all the files
// listed below.
//
// Any JavaScript/Coffee file within this directory, lib/assets/javascripts, or any plugin's
// vendor/assets/javascripts directory can be referenced here using a relative path.
//
// It's not advisable to add code directly here, but if you do, it'll appear at the bottom of the
// compiled file. JavaScript code in this file should be added after the last require_* statement.
//
// Read Sprockets README (https://github.com/rails/sprockets#sprockets-directives) for details
// about supported directives.
//
//= require rails-ujs
//= require activestorage
//= require turbolinks
//= require_tree .
```

AULA 45 - ASSET PIPELINE 2/2

rails g scaffold Customer name:string address:string

rake db:migrate

```
C:\Sites\ror-jp-p4>rake db:migrate
== 20191028192835 CreateCustomers: migrating =====
-- create_table(:customers)
   -> 1.4492s
== 20191028192835 CreateCustomers: migrated (1.4506s) =====
```

Assets para controllers específicos

params[:controller]

app/assets/javascripts/application.js

```
// This is a manifest file that'll be compiled into application.js, which will include all the files
// listed below.
//
// Any JavaScript/Coffee file within this directory, lib/assets/javascripts, or any plugin's
// vendor/assets/javascripts directory can be referenced here using a relative path.
//
// It's not advisable to add code directly here, but if you do, it'll appear at the bottom of the
// compiled file. JavaScript code in this file should be added after the last require_* statement.
//
// Read Sprockets README (https://github.com/rails/sprockets#sprockets-directives) for details
// about supported directives.
//
//= require rails-ujs
//= require activestorage
//= require turbolinks
//= require_tree
```

app/assets/stylesheets/application.css

```
/*
 * This is a manifest file that'll be compiled into application.css, which will include all the files
 * listed below.
 *
 * Any CSS and SCSS file within this directory, lib/assets/stylesheets, or any plugin's
 * vendor/assets/stylesheets directory can be referenced here using a relative path.
 *
 * You're free to add application-wide styles to this file and they'll appear at the bottom of the
 * compiled file so the styles you add here take precedence over styles defined in any other
 * CSS/SCSS
 * files in this directory. Styles in this file should be added after the last require_* statement.
 * It is generally better to create a new file per style scope.
 *
*= require_tree
*= require_self
*/
```

Precompiler Assets Adicionais

Rails.application.config.assets.precompile += %w(search.js)

config/initializers/assets.rb

```
# Be sure to restart your server when you modify this file.

# Version of your assets, change this if you want to expire all your assets.
Rails.application.config.assets.version = '1.0'

# Add additional assets to the asset load path.
# Rails.application.config.assets.paths << Emoji.images_path
# Add Yarn node_modules folder to the asset load path.
Rails.application.config.assets.paths << Rails.root.join('node_modules')

# Precompile additional assets.
# application.js, application.css, and all non-JS/CSS in the app/assets
# folder are already added.
# Rails.application.config.assets.precompile += %w( admin.js admin.css )
Rails.application.config.assets.precompile += %w( customers.coffee
          customers.scss
          users.coffee
          users.scss )
```


content_for

```
<%= yield :head %>

<% content_for :head do %>
  ....
<% end %>
```

app/views/layouts/application.html.erb

```
<!DOCTYPE html>
<html>
  <head>
    <title>RorJpP4</title>
    <%= csrf_meta_tags %>
    <%= csp_meta_tag %>

    <%= stylesheet_link_tag 'application', media: 'all', 'data-turbolinks-track': 'reload' %>
    <%= javascript_include_tag 'application', 'data-turbolinks-track': 'reload' %>

    <%= yield :head %>

  </head>

  <body>
    <%= yield %>
  </body>
</html>
```

app/views/customers/index.html.erb

```
<% content_for :head do %>
  <%= stylesheet_link_tag params[:controller] %>
  <%= javascript_include_tag params[:controller] %>
<% end %>

<p id="notice"><%= notice %></p>

<h1>Customers</h1>

<table>
  <thead>
    <tr>
      <th>Name</th>
      <th>Address</th>
      <th colspan="3"></th>
    </tr>
  </thead>

  <tbody>
    <% @customers.each do |customer| %>
      <tr>
        <td><%= customer.name %></td>
        <td><%= customer.address %></td>
        <td><%= link_to 'Show', customer %></td>
        <td><%= link_to 'Edit', edit_customer_path(customer) %></td>
        <td><%= link_to 'Destroy', customer, method: :delete, data: { confirm: 'Are you sure?' } %></td>
      </tr>
    <% end %>
  </tbody>
</table>

<br>

<%= link_to 'New Customer', new_customer_path %>
```

app/views/users/index.html.erb

```
<% content_for :head do %>
  <%= stylesheet_link_tag params[:controller] %>
  <%= javascript_include_tag params[:controller] %>
<% end %>

<p id="notice"><%= notice %></p>

<h1>Users</h1>

<table>
  <thead>
    <tr>
      <th>Name</th>
      <th>Email</th>
      <th colspan="3"></th>
    </tr>
  </thead>

  <tbody>
    <% @users.each do |user| %>
      <tr>
        <td><%= user.name %></td>
        <td><%= user.email %></td>
        <td><%= link_to 'Show', user %></td>
        <td><%= link_to 'Edit', edit_user_path(user) %></td>
        <td><%= link_to 'Destroy', user, method: :delete, data: { confirm: 'Are you sure?' } %></td>
      </tr>
    <% end %>
  </tbody>
</table>

<br>

<%= link_to 'New User', new_user_path %>
```

AULA 46 - AJAX

- Copie o arquivo **jquery.js** para dentro da pasta **app/assets/javascripts**.

app/assets/javascripts/application.js

```
// This is a manifest file that'll be compiled into application.js, which will include all the files
// listed below.
//
// Any JavaScript/Coffee file within this directory, lib/assets/javascripts, or any plugin's
// vendor/assets/javascripts directory can be referenced here using a relative path.
//
// It's not advisable to add code directly here, but if you do, it'll appear at the bottom of the
// compiled file. JavaScript code in this file should be added after the last require_* statement.
//
// Read Sprockets README (https://github.com/rails/sprockets#sprockets-directives) for details
// about supported directives.
//
//= require jquery
//= require rails-ujs
//= require activestorage
//= require turbolinks
```

app/views/users/new.html.erb

```
<div class="message">
  <h1 style="color:green;">Teste!</h1>
</div>

<h1>New User</h1>

<%= render 'form', user: @user %>

<%= link_to 'Back', users_path %>
```

Testes com o jquery

The image displays three sequential screenshots of a web browser window titled 'RorJpP4' at the URL 'localhost:3000/users/new'. The browser's developer console is open, showing jQuery commands and their outputs.

First Screenshot: The page shows a 'New User' form with input fields for 'Name' and 'Email', and a 'Create User' button. The console shows the command `$('.message h1').remove();` and its output: `Object { 0: h1, length: 1, prevObject: Object(1) }`.

Second Screenshot: The page now displays a green heading 'Teste!' above the 'New User' form. The console shows the command `$('.message').html("<h1 style='color:green;'>Teste!</h1>");` and its output: `Object { 0: div.message, length: 1, prevObject: Object(1) }`.

Third Screenshot: The form inputs are filled with 'Jackson' and 'jackson@gmail.com'. The console shows two commands: `$('#user_name').val("Jackson");` and `$('#user_email').val("jackson@gmail.com");`, both returning `Object [input#user_name]` and `Object [input#user_email]` respectively.

Rails + Ajax

app/views/users/_form.html.erb

```
<%= form_with(model: user, remote: true, id: "form_user", local: true) do |form| %>
  <% if user.errors.any? %>
    <div id="error_explanation">
      <h2><%= pluralize(user.errors.count, "error") %> prohibited this user from being saved:</h2>

      <ul>
        <% user.errors.full_messages.each do |message| %>
          <li><%= message %></li>
        <% end %>
      </ul>
    </div>
  <% end %>

  <div class="field">
    <%= form.label :name %>
    <%= form.text_field :name %>
  </div>

  <div class="field">
    <%= form.label :email %>
    <%= form.text_field :email %>
  </div>

  <div class="actions">
    <%= form.submit %>
  </div>
<% end %>
```

app/controllers/users_controller.rb

```
class UsersController < ApplicationController
  before_action :set_user, only: [:show, :edit, :update, :destroy]

  # GET /users
  # GET /users.json
  def index
    @users = User.all
  end

  # GET /users/1
  # GET /users/1.json
  def show
  end

  # GET /users/new
  def new
    @user = User.new
  end

  # GET /users/1/edit
  def edit
  end

  # POST /users
  # POST /users.json
  def create
    @user = User.new(user_params)

    respond_to do |format|
      if @user.save
        format.html { redirect_to @user, notice: 'User was successfully created.' }
        format.json { render :show, status: :created, location: @user }
        format.js # views/users/create.js.erb
      else
        format.html { render :new }
        format.json { render json: @user.errors, status: :unprocessable_entity }
        format.js # views/users/create.js.erb
      end
    end
  end

  # PATCH/PUT /users/1
  # PATCH/PUT /users/1.json
  def update
    respond_to do |format|
      if @user.update(user_params)
        format.html { redirect_to @user, notice: 'User was successfully updated.' }
        format.json { render :show, status: :ok, location: @user }
      end
    end
  end
end
```

```

    else
      format.html { render :edit }
      format.json { render json: @user.errors, status: :unprocessable_entity }
    end
  end
end

# DELETE /users/1
# DELETE /users/1.json
def destroy
  @user.destroy
  respond_to do |format|
    format.html { redirect_to users_url, notice: 'User was successfully destroyed.' }
    format.json { head :no_content }
  end
end

private
# Use callbacks to share common setup or constraints between actions.
def set_user
  @user = User.find(params[:id])
end

# Never trust parameters from the scary internet, only allow the white list through.
def user_params
  params.require(:user).permit(:name, :email)
end
end

```

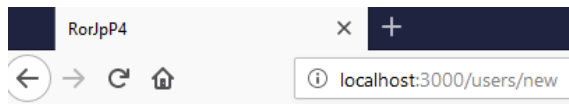
app/views/users/create.js.erb

```

var qtdErrors = <%= @user.errors.count %>;

if (qtdErrors > 0) {
  $(".message").html("<h1>Ocorreu um erro!</h1>").css({color: "red"});
} else {
  $(".message").html("<h1>Cadastrado com sucesso!</h1>").css({color: "green"});
  $("#user_name").val("");
  $("#user_email").val("");
}
setTimeout(function(){ $(".message h1").remove() }, 5000);

```

Ocorreu um erro!

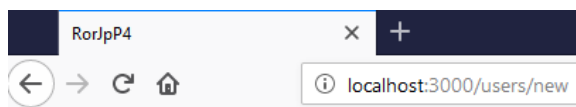
New User

Name

Email

[Create User](#)

[Back](#)



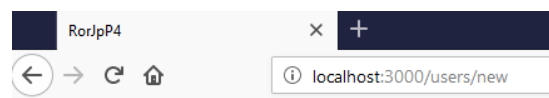
New User

Name

Email

[Create User](#)

[Back](#)



Cadastrado com sucesso!

New User

Name

Email

[Create User](#)

[Back](#)

AULA 47 - NOTIFY.JS + RAILS

<https://notifyjs.jpillora.com/>

- Baixar **notify.js** e copiá-lo para a pasta **app/assets/javascripts**.

config/initializers/assets.rb

```
# Be sure to restart your server when you modify this file.

# Version of your assets, change this if you want to expire all your assets.
Rails.application.config.assets.version = '1.0'

# Add additional assets to the asset load path.
# Rails.application.config.assets.paths << Emoji.images_path
# Add Yarn node_modules folder to the asset load path.
Rails.application.config.assets.paths << Rails.root.join('node_modules')

# Precompile additional assets.
# application.js, application.css, and all non-JS/CSS in the app/assets
# folder are already added.
# Rails.application.config.assets.precompile += %w( admin.js admin.css )
Rails.application.config.assets.precompile += %w( customers.coffee
          customers.scss
          users.coffee
          users.scss
          jquery.js
          notify.js )
```

app/views/users/new.html.erb

```
<% content_for :head do %>
  <%= javascript_include_tag 'notify' %>
<% end %>

<h1>New User</h1>

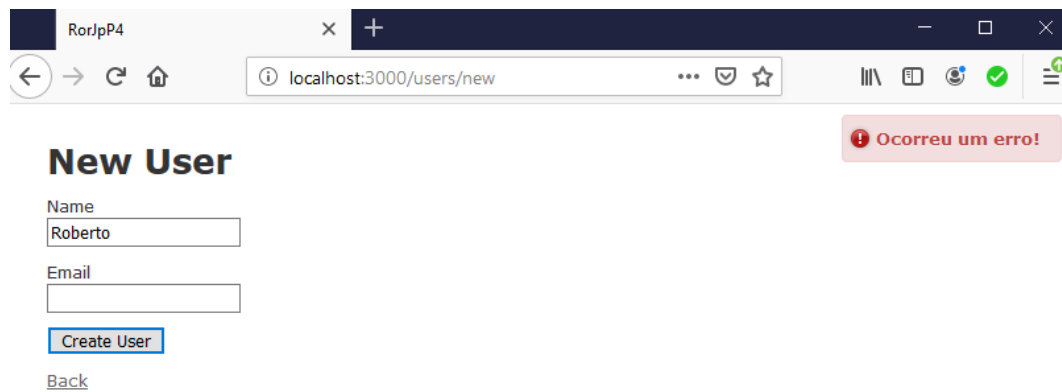
<%= render 'form', user: @user %>

<%= link_to 'Back', users_path %>
```

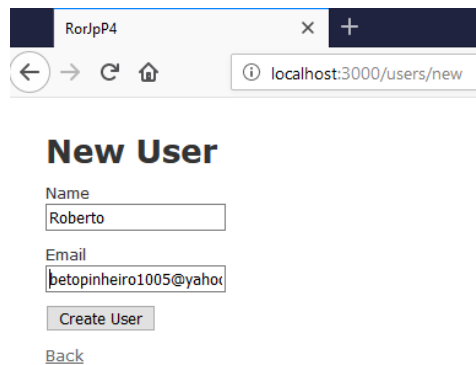
app/views/users/create.js.erb

```
var qtdErrors = <%= @user.errors.count %>;

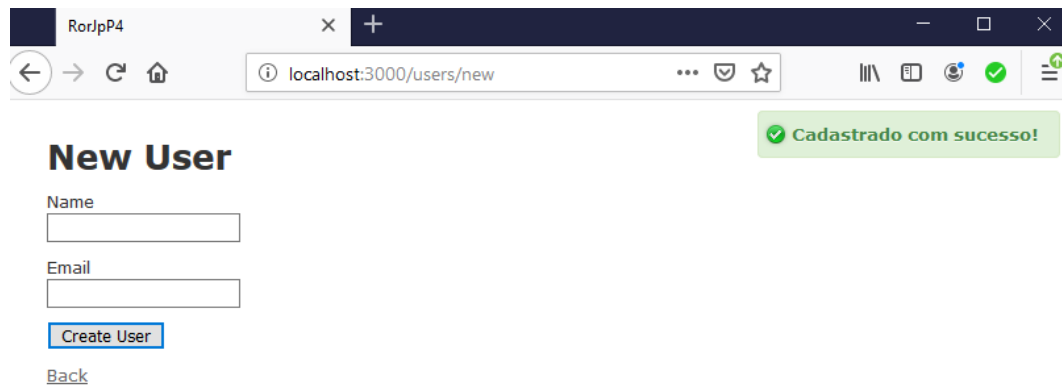
if (qtdErrors > 0) {
  $.notify("Ocorreu um erro!", "error");
} else {
  $.notify("Cadastrado com sucesso!", "success");
  $("#user_name").val("");
  $("#user_email").val("");
}
```



A browser window titled 'RorJpP4' shows the 'New User' form at 'localhost:3000/users/new'. The form has two input fields: 'Name' with the value 'Roberto' and 'Email' which is empty. Below the fields is a blue 'Create User' button and a 'Back' link. A red error notification box in the top right corner displays 'Ocorreu um erro!' with a red exclamation mark icon.



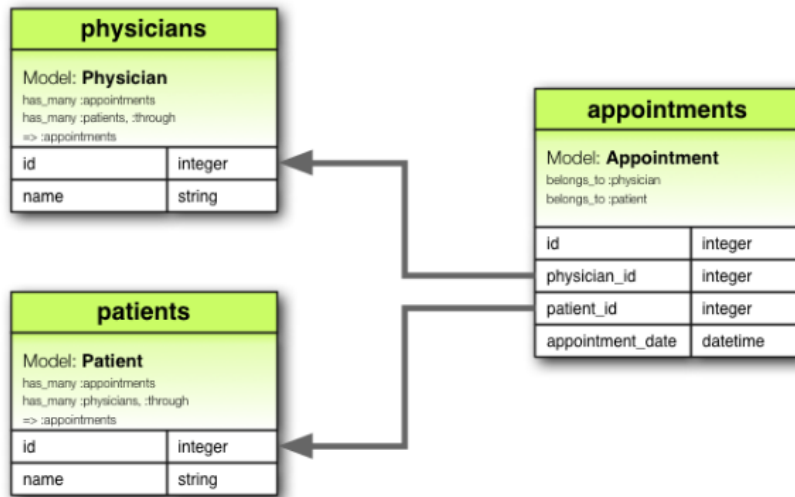
The same browser window shows the 'New User' form. The 'Name' field contains 'Roberto' and the 'Email' field contains 'betopinheiro1005@yahoo'. The 'Create User' button is now greyed out (disabled). The 'Back' link remains at the bottom.



The browser window shows the 'New User' form after successful submission. The 'Name' and 'Email' fields are now empty. The 'Create User' button is blue and active. A green success notification box in the top right corner displays 'Cadastrado com sucesso!' with a green checkmark icon. The 'Back' link is still present at the bottom.

AULA 48 - HAS MANY THROUGH + COCOON GEM

https://guides.rubyonrails.org/association_basics.html#the-has-many-through-association



The screenshot shows a web form for adding a consultation. At the top, there is a text input field labeled "Nome do Médico" with the value "Fulano de Tal". Below this is a link "Adicionar Consulta" with a circular arrow icon. The main form area contains two input fields: "Horário" with the value "12/12/2016 - 18:00" and "Nome do Paciente" with the value "Paciente tal". A "Remover" link is next to the patient name field. At the bottom of the form is a "Salvar" button.

Gerando os scaffolds

```
rails generate scaffold Physician name:string
```

```
rails generate scaffold Appointment physician:references patient:references appointment_date:datetime
```

```
rails generate scaffold Patient name:string
```

```
rake db:migrate
```

```
C:\Sites\ror-jp-p4>rake db:migrate
== 20191029165440 CreatePhysicians: migrating =====
-- create_table(:physicians)
--> 0.3171s
== 20191029165440 CreatePhysicians: migrated (0.3190s) =====

== 20191029165531 CreatePatients: migrating =====
-- create_table(:patients)
--> 0.2755s
== 20191029165531 CreatePatients: migrated (0.2777s) =====

== 20191029165615 CreateAppointments: migrating =====
-- create_table(:appointments)
--> 0.6068s
== 20191029165615 CreateAppointments: migrated (0.6095s) =====
```

app/models/patient.rb

```
class Patient < ActiveRecord::Base
  has_many :appointments
  has_many :physicians, through: :appointments
end
```

app/models/physician.rb

```
class Physician < ActiveRecord::Base
  has_many :appointments
  has_many :patients, through: :appointments
end
```

app/models/appointment.rb

```
class Appointment < ApplicationRecord
  belongs_to :physician
  belongs_to :patient
end
```

Nested Attributes

app/models/appointment.rb

```
class Appointment < ApplicationRecord
  belongs_to :physician
  belongs_to :patient

  accepts_nested_attributes_for :patient, reject_if: :all_blank, allow_destroy: true
end
```

app/models/physician.rb

```
class Physician < ActiveRecord::Base
  has_many :appointments
  has_many :patients, through: :appointments

  accepts_nested_attributes_for :appointments, reject_if: :all_blank, allow_destroy: true
end
```

Strong Parameters

```
def physician_params
  params.require(:physician).permit(:name, :appointments_attributes => [:id, :appointment_date,
:physician_id, :patient_id, :_destroy, :patient_attributes => [:id, :name]])
end
```

app/controllers/physicians_controller.rb

```
class PhysiciansController < ApplicationController
  before_action :set_physician, only: [:show, :edit, :update, :destroy]

  # GET /physicians
  # GET /physicians.json
  def index
    @physicians = Physician.all
  end

  # GET /physicians/1
  # GET /physicians/1.json
  def show
  end

  # GET /physicians/new
  def new
    @physician = Physician.new
  end

  # GET /physicians/1/edit
  def edit
  end

  # POST /physicians
  # POST /physicians.json
  def create
    @physician = Physician.new(physician_params)

    respond_to do |format|
      if @physician.save
        format.html { redirect_to @physician, notice: 'Physician was successfully created.' }
        format.json { render :show, status: :created, location: @physician }
      else
        format.html { render :new }
        format.json { render json: @physician.errors, status: :unprocessable_entity }
      end
    end
  end

  # PATCH/PUT /physicians/1
```

```

# PATCH/PUT /physicians/1.json
def update
  respond_to do |format|
    if @physician.update(physician_params)
      format.html { redirect_to @physician, notice: 'Physician was successfully updated.' }
      format.json { render :show, status: :ok, location: @physician }
    else
      format.html { render :edit }
      format.json { render json: @physician.errors, status: :unprocessable_entity }
    end
  end
end

# DELETE /physicians/1
# DELETE /physicians/1.json
def destroy
  @physician.destroy
  respond_to do |format|
    format.html { redirect_to physicians_url, notice: 'Physician was successfully destroyed.' }
    format.json { head :no_content }
  end
end

private
# Use callbacks to share common setup or constraints between actions.
def set_physician
  @physician = Physician.find(params[:id])
end

# Never trust parameters from the scary internet, only allow the white list through.
def physician_params
  params.require(:physician).permit(:name, :appointments_attributes => [:id, :appointment_date,
:physician_id, :patient_id, :_destroy, :patient_attributes => [:id, :name]])
end
end

```


COCOON

<https://github.com/nathanvda/cocoon>

Gemfile

```
source 'https://rubygems.org'
git_source(:github) { |repo| "https://github.com/#{repo}.git" }

ruby '2.5.1'

gem 'rails', '~> 5.2.3'
# Use mysql as the database for Active Record
gem 'mysql2', '>= 0.4.4', '< 0.6.0'
# Use Puma as the app server
gem 'puma', '~> 3.11'
# Use SCSS for stylesheets
gem 'sass-rails', '~> 5.0'
# Use Uglifier as compressor for JavaScript assets
gem 'uglifier', '>= 1.3.0'
# See https://github.com/rails/execjs#readme for more supported runtimes
gem 'duktape'
# Use CoffeeScript for .coffee assets and views
gem 'coffee-rails', '~> 4.2'
# Turbolinks makes navigating your web application faster. Read more:
# https://github.com/turbolinks/turbolinks
gem 'turbolinks', '~> 5'
# Build JSON APIs with ease. Read more: https://github.com/rails/jbuilder
gem 'jbuilder', '~> 2.5'
# Use Redis adapter to run Action Cable in production
# gem 'redis', '~> 4.0'
# Use ActiveModel has_secure_password
# gem 'bcrypt', '~> 3.1.7'

# Use ActiveSupport variant
# gem 'mini_magick', '~> 4.8'

# Use Capistrano for deployment
# gem 'capistrano-rails', group: :development

# Reduces boot times through caching; required in config/boot.rb
gem 'bootsnap', '>= 1.1.0', require: false
# Cocoon
gem "cocoon"

group :development, :test do
  # Call 'byebug' anywhere in the code to stop execution and get a debugger console
  gem 'byebug', platforms: [:mri, :mingw, :x64_mingw]
```

```
gem 'rspec-rails', '~> 3.0'
gem 'capybara'
end

group :development do
  # Access an interactive console on exception pages or by calling 'console' anywhere in the code.
  gem 'web-console', '>= 3.3.0'
end

# Windows does not include zoneinfo files, so bundle the tzinfo-data gem
gem 'tzinfo-data', platforms: [:mingw, :mswin, :x64_mingw, :jruby]
```

bundle install

Rails 3.1+/Rails 4/Rails 5

Add the following to application.js so it compiles to the asset pipeline:

//= require cocoon

app/assets/javascripts/application.js

```
// This is a manifest file that'll be compiled into application.js, which will include all the files
// listed below.
//
// Any JavaScript/Coffee file within this directory, lib/assets/javascripts, or any plugin's
// vendor/assets/javascripts directory can be referenced here using a relative path.
//
// It's not advisable to add code directly here, but if you do, it'll appear at the bottom of the
// compiled file. JavaScript code in this file should be added after the last require_* statement.
//
// Read Sprockets README (https://github.com/rails/sprockets#sprockets-directives) for details
// about supported directives.
//
//= require jquery
//= require rails-ujs
//= require activestorage
//= require turbolinks
//= require cocoon
```

HELPERS

link_to_add_association

This function adds a link to your markup that, when clicked, dynamically adds a new partial form for the given association.

link_to_remove_association

This function will add a link to your markup that, when clicked, dynamically removes the surrounding partial form. This should be placed inside the partial `<association-object-singular>_fields`.

FORMULÁRIO PHYSICIAN

...

```
<%= link_to_add_association 'add appointment', f, :appointments, 'data-association-insertion-node'
=> "#appointments-patient ol", 'data-association-insertion-method' => "append", :wrap_object =>
Proc.new {|appointment| appointment.build_patient; appointment } %>
```

```
<hr/>
```

```
<fieldset id="appointments-patient">
  <ol>
    <%= f.fields_for :appointments do |appointment| %>
      <%= render partial: "appointment_fields", locals: { f:appointment } %>
    <% end %>
  </ol>
</fieldset>
```

`_appointment_fields.html.erb`

```
<li class="control-group nested-fields">
  <div class="controls">
    <%= f.datetime_select :appointment_date %>>
    <%= f.fields_for :patient do |appointment_patient| %>
      <%= appointment_patient.text_field :name %>
    <% end %>
    <%= link_to_remove_association "remove", f %>
  </div>
</li>
```

app/views/physicians/_form.html.erb

```
<%= form_for @physician do |f| %>
  <% if @physician.errors.any? %>
    <div id="error_explanation">
      <h2><%= pluralize(@physician.errors.count, "error") %> prohibited this physician from being saved:</h2>

      <ul>
        <% @physician.errors.full_messages.each do |message| %>
          <li><%= message %></li>
        <% end %>
      </ul>
    </div>
  <% end %>

  <div class="field">
    <%= f.label :name %>
    <%= f.text_field :name %>

    <br><br>

    <%= link_to_add_association 'add appointment', f,
      :appointments, 'data-association-insertion-node' => "#appointments-patient ol",
      'data-association-insertion-method' => "append",
      :wrap_object => Proc.new { |appointment| appointment.build_patient; appointment },
      class: "btn btn-primary"

    %>

    <hr/>

    <fieldset id="appointments-patient">
      <ol>
        <%= f.fields_for :appointments do |appointment| %>
          <%= render partial: "appointment_fields", locals: { f:appointment } %>
        <% end %>
      </ol>
    </fieldset>
  </div>

  <div class="actions">
    <%= f.submit class: "btn btn-success" %>
  </div>
<% end %>
```

app/views/physicians/_appointment_fields.html.erb

```
<li class="control-group nested-fields">
  <div class="controls">
    <%= f.datetime_select :appointment_date %>>
    <%= f.fields_for :patient do |appointment_patient| %>
      <%= appointment_patient.text_field :name %>
    <% end %>
    <%= link_to_remove_association "Remover", f, class: "btn btn-default" %>
  </div>
</li>
```

rake db:migrate

- Baixe e copie o arquivo **bootstrap.css** para a pasta **app/assets/stylesheets**.

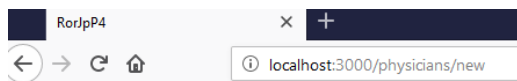
config/initializers/assets.rb

```
# Be sure to restart your server when you modify this file.

# Version of your assets, change this if you want to expire all your assets.
Rails.application.config.assets.version = '1.0'

# Add additional assets to the asset load path.
# Rails.application.config.assets.paths << Emoji.images_path
# Add Yarn node_modules folder to the asset load path.
Rails.application.config.assets.paths << Rails.root.join('node_modules')

# Precompile additional assets.
# application.js, application.css, and all non-JS/CSS in the app/assets
# folder are already added.
# Rails.application.config.assets.precompile += %w( admin.js admin.css )
Rails.application.config.assets.precompile += %w( customers.coffee
  customers.scss
  users.coffee
  users.scss
  jquery.js
  notify.js
  bootstrap.css )
```



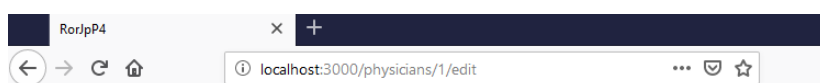
New Physician

Name

add appointment

Create Physician

Back



Editing Physician

Name

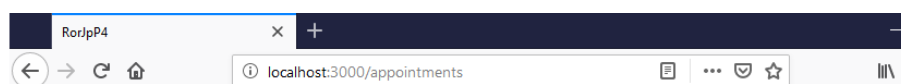
add appointment

1. 2019 October 31 - 16 : 00 > Aurélio Remover
2. 2019 November 12 - 10 : 30 > Marina Remover
3. 2019 November 22 - 17 : 12 > Francisco Remover

Update Physician

Show

Back



Appointments

Physician	Patient	Appointment date			
José Ricardo	Aurélio	2019-10-31 16:00:00 UTC	Show	Edit	Destroy
José Ricardo	Marina	2019-11-12 10:30:00 UTC	Show	Edit	Destroy
José Ricardo	Francisco	2019-11-22 17:12:00 UTC	Show	Edit	Destroy
Bruno	Rita	2019-11-05 09:15:00 UTC	Show	Edit	Destroy
Bruno	Selma	2019-11-16 13:00:00 UTC	Show	Edit	Destroy
Bruno	Anderson	2019-11-27 16:30:00 UTC	Show	Edit	Destroy
Orlando	Leonardo	2019-10-29 22:24:00 UTC	Show	Edit	Destroy

New Appointment

AULA 49 - RAILS ADMIN INTERFACES

RUBY-TOOLBOX

https://www.ruby-toolbox.com/categories/rails_admin_interfaces

ACTIVE SCAFFOLD

https://github.com/activescaffold/active_scaffold

Gemfile

```
source 'https://rubygems.org'
git_source(:github) { |repo| "https://github.com/#{repo}.git" }

ruby '2.5.1'

gem 'rails', '~> 5.2.3'
# Use mysql as the database for Active Record
gem 'mysql2', '>= 0.4.4', '< 0.6.0'
# Use Puma as the app server
gem 'puma', '~> 3.11'
# Use SCSS for stylesheets
gem 'sass-rails', '~> 5.0'
# Use Uglifier as compressor for JavaScript assets
gem 'uglifier', '>= 1.3.0'
# See https://github.com/rails/execjs#readme for more supported runtimes
gem 'duktape'
# Use CoffeeScript for .coffee assets and views
gem 'coffee-rails', '~> 4.2'
# Turbolinks makes navigating your web application faster. Read more:
https://github.com/turbolinks/turbolinks
gem 'turbolinks', '~> 5'
# Build JSON APIs with ease. Read more: https://github.com/rails/jbuilder
gem 'jbuilder', '~> 2.5'
# ActiveSupport
gem 'active_scaffold'
# Use Redis adapter to run Action Cable in production
# gem 'redis', '~> 4.0'
# Use ActiveModel has_secure_password
gem 'bcrypt', '~> 3.1.7'

# Use ActiveSupport variant
gem 'mini_magick', '~> 4.8'
```

```
# Use Capistrano for deployment
# gem 'capistrano-rails', group: :development

# Reduces boot times through caching; required in config/boot.rb
gem 'bootsnap', '>= 1.1.0', require: false
# Cocoon
gem "cocoon"

group :development, :test do
  # Call 'byebug' anywhere in the code to stop execution and get a debugger console
  gem 'byebug', platforms: [:mri, :mingw, :x64_mingw]
  gem 'rspec-rails', '~> 3.0'
  gem 'capybara'
end

group :development do
  # Access an interactive console on exception pages or by calling 'console' anywhere in the code.
  gem 'web-console', '>= 3.3.0'
end

# Windows does not include zoneinfo files, so bundle the tzinfo-data gem
gem 'tzinfo-data', platforms: [:mingw, :mswin, :x64_mingw, :jruby]
```


bundle install

rake db:create

```
C:\Sites\ror-jp-a49>rake db:create
Created database 'ror-jp-a49_development'
Created database 'ror-jp-a49_test'
```

rails g active_scaffold:install

```
C:\Sites\ror-jp-a49>rails g active_scaffold:install
  route concern :active_scaffold, ActiveSupport::Routing::Basic.new(association: true)
  route concern :active_scaffold_association, ActiveSupport::Routing::Association.new
  insert app/assets/javascripts/application.js
  insert app/assets/stylesheets/application.css
```

rails g active_scaffold:resource Category name:string

```
C:\Sites\ror-jp-a49>rails g active_scaffold:resource Category name:string
  invoke active_record
  create db/migrate/20191030134851\_create\_categories.rb
  create app/models/category.rb
  invoke test_unit
  create test/models/category\_test.rb
  create test/fixtures/categories.yml
  route resources :categories, concerns: :active_scaffold
  invoke active_scaffold:controller
  create app/controllers/categories\_controller.rb
  create app/helpers/categories\_helper.rb
  invoke test_unit
  create test/controllers/categories\_controller\_test.rb
  create test/system/categories\_test.rb
  create app/views/categories
```

rake db:migrate

```
C:\Sites\ror-jp-a49>rake db:migrate
== 20191030141752 CreateCategories: migrating =====
-- create_table(:categories)
   -> 0.2884s
== 20191030141752 CreateCategories: migrated (0.2895s) =====
```

ACTIVE ADMIN

Gemfile

```
source 'https://rubygems.org'
git_source(:github) { |repo| "https://github.com/#{repo}.git" }

ruby '2.5.1'

gem 'rails', '~> 5.2.1'
# Use mysql as the database for Active Record
gem 'mysql2', '>= 0.4.4', '< 0.6.0'
# Use Puma as the app server
gem 'puma', '~> 3.11'
# Use SCSS for stylesheets
gem 'sass-rails', '~> 5.0'
# Use Uglifier as compressor for JavaScript assets
gem 'uglifier', '>= 1.3.0'
# See https://github.com/rails/execjs#readme for more supported runtimes
gem 'duktape'
# Use CoffeeScript for .coffee assets and views
gem 'coffee-rails', '~> 4.2'
# Turbolinks makes navigating your web application faster. Read more:
https://github.com/turbolinks/turbolinks
gem 'turbolinks', '~> 5'
# Build JSON APIs with ease. Read more: https://github.com/rails/jbuilder
gem 'jbuilder', '~> 2.5'
gem 'activeadmin'
gem 'devise'
# Use Redis adapter to run Action Cable in production
# gem 'redis', '~> 4.0'
# Use ActiveModel has_secure_password
# gem 'bcrypt', '~> 3.1.7'

# Use ActiveSupport variant
# gem 'mini_magick', '~> 4.8'

# Use Capistrano for deployment
# gem 'capistrano-rails', group: :development

# Reduces boot times through caching; required in config/boot.rb
gem 'bootsnap', '>= 1.1.0', require: false
# Cocoon
gem "cocoon"

group :development, :test do
  # Call 'byebug' anywhere in the code to stop execution and get a debugger console
  gem 'byebug', platforms: [:mri, :mingw, :x64_mingw]
  gem 'rspec-rails', '~> 3.0'
```

```
gem 'capybara'
end

group :development do
  # Access an interactive console on exception pages or by calling 'console' anywhere in the code.
  gem 'web-console', '>= 3.3.0'
end

# Windows does not include zoneinfo files, so bundle the tzinfo-data gem
gem 'tzinfo-data', platforms: [:mingw, :mswin, :x64_mingw, :jruby]
```

bundle install

rails g active_admin:install

rake db:migrate

rake db:seed

db/seeds.rb

```
# This file should contain all the record creation needed to seed the database with its default values.
# The data can then be loaded with the rails db:seed command (or created alongside the database
# with db:setup).
#
# Examples:
#
# movies = Movie.create([{ name: 'Star Wars' }, { name: 'Lord of the Rings' }])
# Character.create(name: 'Luke', movie: movies.first)
AdminUser.create!(email: 'admin@example.com', password: 'password', password_confirmation:
'password') if Rails.env.development?
```

rails generate scaffold Customer name:string address:string

rake db:migrate

rails generate active_admin:resource Customer

```
(C:\Sites\ror-jp-p4>rails generate active_admin:resource Customer
create app/admin/customers.rb
```

Visit <http://localhost:3000/admin> and log in using:

User: `admin@example.com`

Password: `password`

You need to sign in or sign up before continuing.

Ror Jp P4 Login

Email*:

Password*:

☐ Remember me

Login [Forgot your password?](#)

ADMIN / **Admin Users** New Admin User

Batch Actions ▾

<input type="checkbox"/>	Id	Email	Current Sign In At	Sign In Count	Created At	
<input type="checkbox"/>	2	jackson@gmail.com			October 30, 2019 22:30	View Edit Delete
<input type="checkbox"/>	1	admin@example.com			October 30, 2019 19:17	View Edit Delete

Download: [CSV](#) [XML](#) [JSON](#) Displaying all 2 Admin Users

Powered by [Active Admin](#) 2.4.0

Filters

EMAIL

CREATED AT
From To

Filter Clear Filters

ADMIN / **Customers** New Customer

Batch Actions ▾

There are no Customers yet. [Create one](#)

Powered by [Active Admin](#) 2.4.0

Filters

NAME

ADDRESS

CREATED AT
From To

UPDATED AT
From To

Filter Clear Filters