

Curso Ruby on Rails

Codifike (Fabrício Mattos)

https://www.youtube.com/watch?v=2SEKJdKzwNU&list=PLFeyfVYazTkJN6uM5opCfSN_xjxrMybXV

Resumo do curso feito por Roberto Pinheiro

AULA 02 - CRIANDO A APLICAÇÃO

INSTALANDO O RUBY E O RAILS

- Instale o Ruby v. 2.5.1.2 com msys2 64bit e Rails v. 5.2.1
- Baixar o Ruby em:

<https://rubyinstaller.org/downloads/archives/>

`ruby -v`

```
C:\Users\pinhe>ruby -v
ruby 2.5.1p57 (2018-03-29 revision 63029) [x64-mingw32]
```

`gem -v`

```
C:\Users\pinhe>gem -v
2.7.6
```

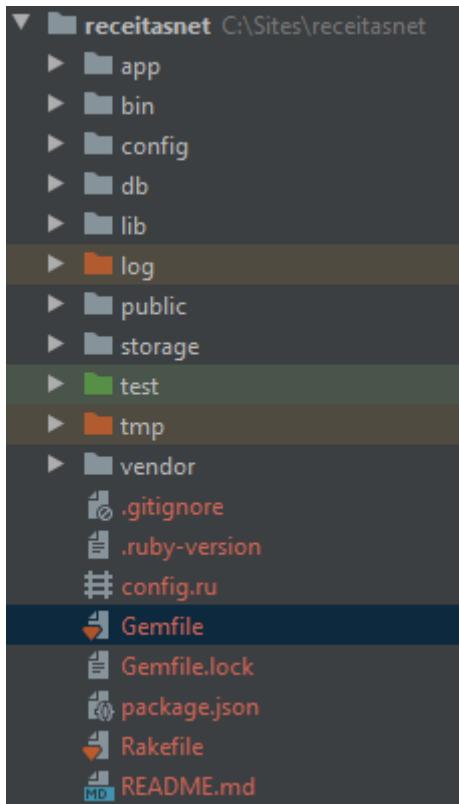
`gem install rails -v 5.2.1`

`rails -v`

```
C:\Users\pinhe>rails -v
Rails 5.2.1
```

CRIANDO A APLICAÇÃO

```
rails new receitasnet --database=mysql
```



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', '~> 5.2.1'  
# 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'

# Use Redis adapter to run Action Cable in production
# gem 'redis', '~> 4.0'

# Use ActiveModel has_secure_password
# gem 'bcrypt', '~> 3.1.7'

# Use ActiveStorage 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]
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

group :test do
  # Adds support for Capybara system testing and selenium driver
  gem 'capybara', '>= 2.15'
  gem 'selenium-webdriver'
  # Easy installation and use of chromedriver to run system tests with Chrome
  gem 'chromedriver-helper'
end

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

```

Para atualizar, no terminal, entre com o comando:

```
bundle install
```

```
bundle check
```

```
C:\Sites\receitasnet>bundle check  
The Gemfile's dependencies are satisfied
```

criando o banco de dados

config/database.yml

```
# MySQL. Versions 5.1.10 and up are supported.  
#  
# Install the MySQL driver  
# gem install mysql2  
#  
# Ensure the MySQL gem is defined in your Gemfile  
# gem 'mysql2'  
#  
# And be sure to use new-style password hashing:  
# https://dev.mysql.com/doc/refman/5.7/en/password-hashing.html  
#  
default: &default  
adapter: mysql2  
encoding: utf8  
pool: <%= ENV.fetch("RAILS_MAX_THREADS") { 5 } %>  
username: root  
password:  
host: localhost  
  
development:  
<<: *default  
database: receitasnet_development  
  
# Warning: The database defined as "test" will be erased and  
# re-generated from your development database when you run "rake".  
# Do not set this db to the same as development or production.  
test:  
<<: *default  
database: receitasnet_test  
  
# As with config/secrets.yml, you never want to store sensitive information,  
# like your database password, in your source code. If your source code is  
# ever seen by anyone, they now have access to your database.
```

```
#  
# Instead, provide the password as a unix environment variable when you boot  
# the app. Read http://guides.rubyonrails.org/configuring.html#configuring-a-database  
# for a full rundown on how to provide these environment variables in a  
# production deployment.  
#  
# On Heroku and other platform providers, you may have a full connection URL  
# available as an environment variable. For example:  
#  
# DATABASE_URL="mysql2://myuser:mypassword@localhost/somedatabase"  
#  
# You can use this database configuration with:  
#  
# production:  
#   url: <%= ENV['DATABASE_URL'] %>  
#  
production:  
<<: *default  
database: receitasnet_production  
username: receitasnet  
password: <%= ENV['RECEITASNET_DATABASE_PASSWORD'] %>
```

- Inicie o servidor Apache e o MySQL.

- Entre com o comando:

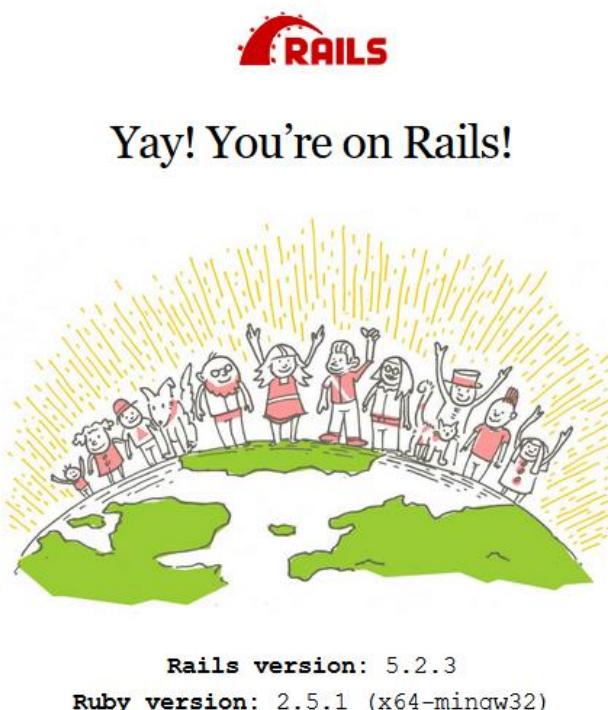
rails db:create

```
C:\Sites\receitasnet>rails db:create  
Created database 'receitasnet_development'  
Created database 'receitasnet_test'
```

INICIANDO O SERVIDOR

```
rails s
```

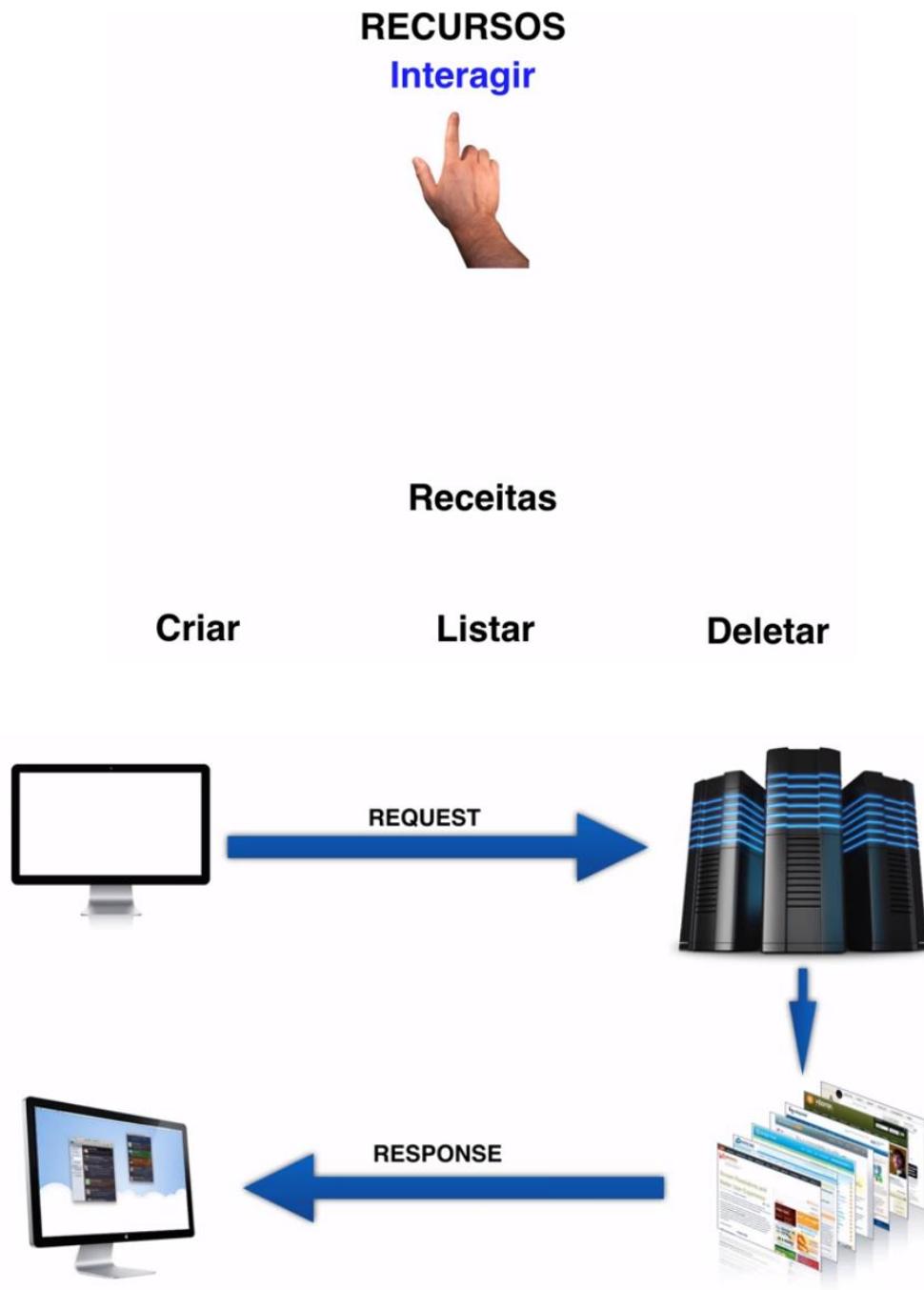
```
C:\Sites\receitasnet>rails s
=> Booting Puma
=> Rails 5.2.3 application starting in development
=> Run `rails server -h` for more startup options
*** SIGUSR2 not implemented, signal based restart unavailable!
*** SIGUSR1 not implemented, signal based restart unavailable!
*** SIGHUP not implemented, signal based logs reopening unavailable!
Puma starting in single mode...
* Version 3.12.1 (ruby 2.5.1-p57), codename: Llamas in Pajamas
* Min threads: 5, max threads: 5
* Environment: development
* Listening on tcp://localhost:3000
Use Ctrl-C to stop
```



AULA 03 – ENTENDENDO V E C DO MVC

RECURSOS

Recurso é tudo aquilo que conseguimos interagir dentro da nossa aplicação.



ARQUITETURA REST

O Rails trabalha com a arquitetura REST.

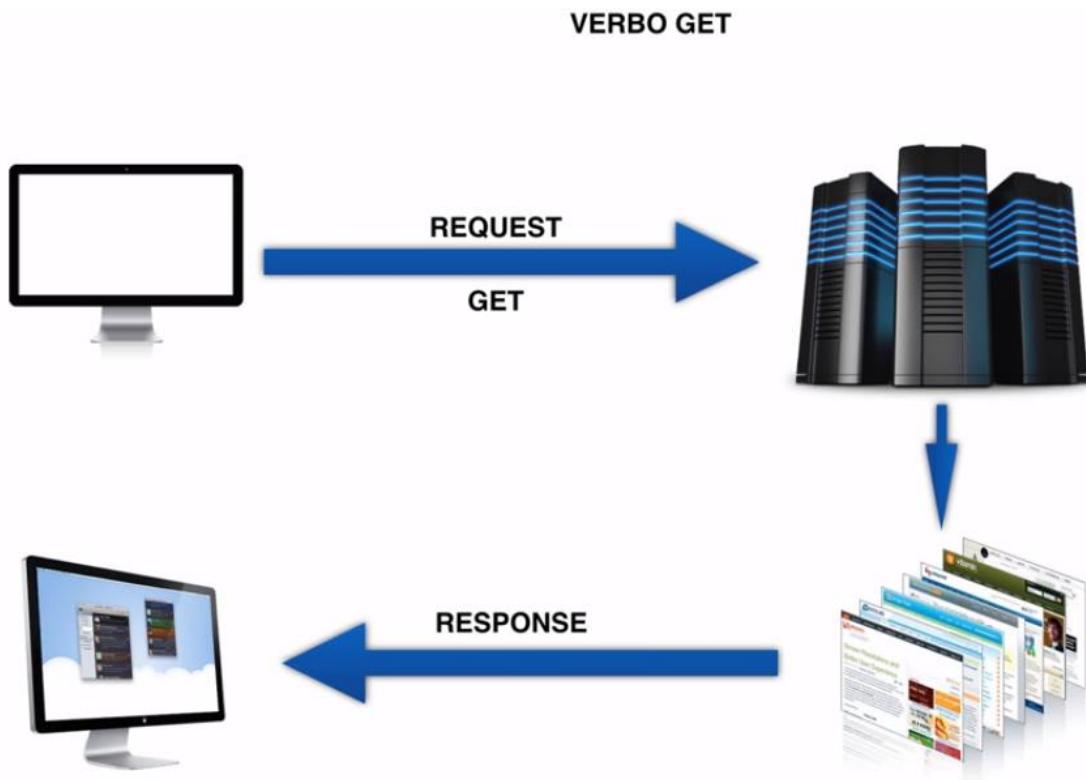
VERBOS HTTP

O protocolo HTTP básico define algumas operações:

- GET
- POST
- PATCH
- DELETE

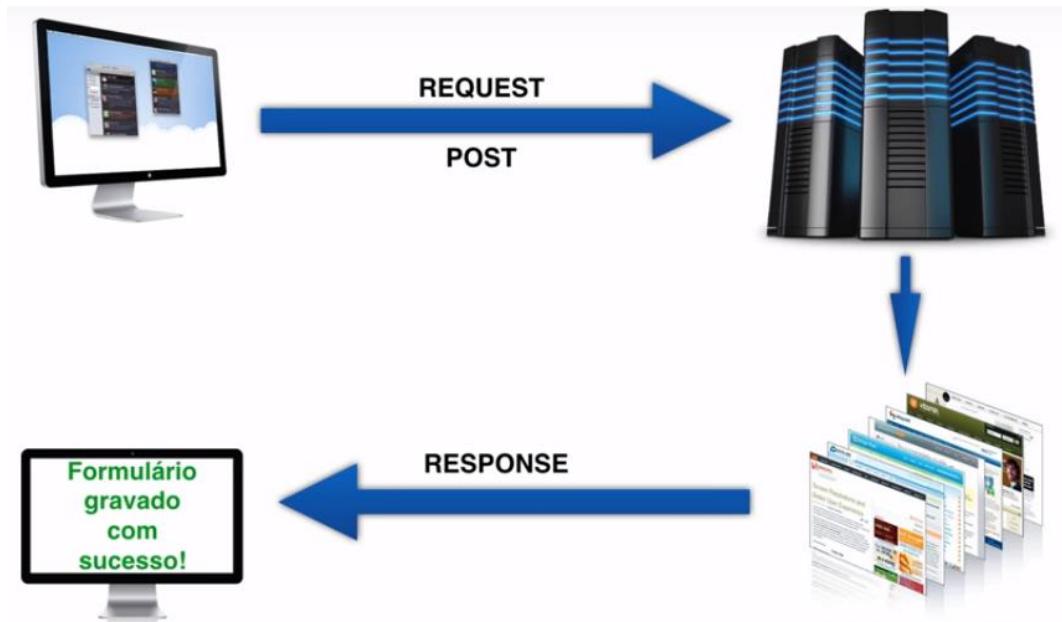
GET

É a operação mais comum. O verbo GET é utilizado para leitura de dados na WEB.



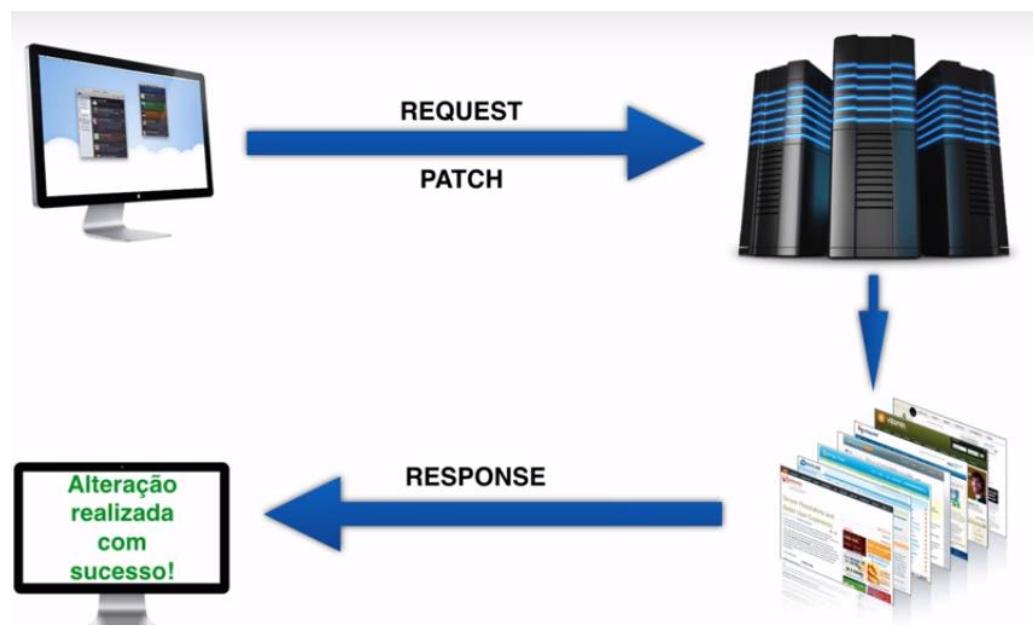
POST

É o pedido enviado pelo navegador quando se envia, por exemplo, um formulário. As operações POST são utilizadas para criar coisas.



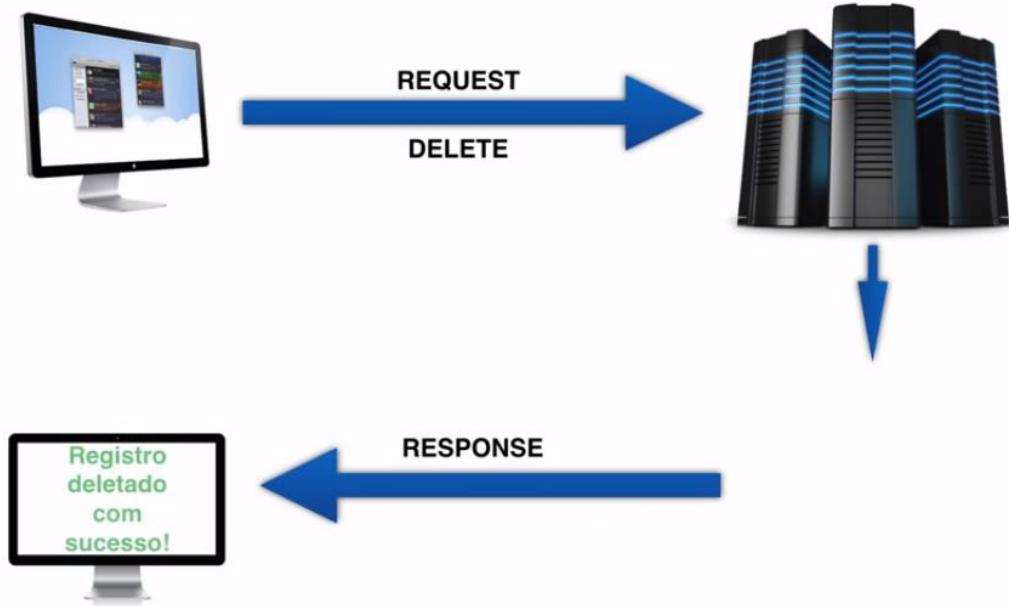
PATCH

É utilizado para atualizar informações.



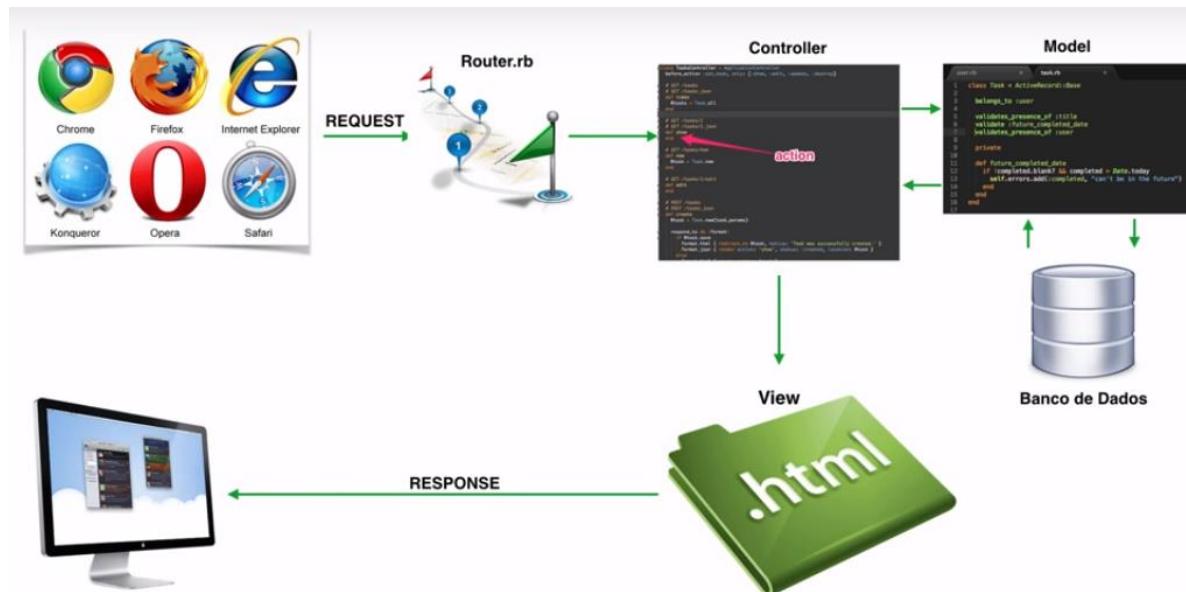
DELETE

É utilizado para destruir ou eliminar informações no servidor.



CONFIGURANDO O ARQUIVO DE ROTAS

Rota é o caminho que a aplicação tem que tomar para chegar num determinado recurso.



config/routes.rb

```
Rails.application.routes.draw do
  resources :recipes
end
```

Listando as rotas criadas

rake routes

```
C:\Sites\receitasnet>rake routes
      Prefix Verb    URI Pattern                               Controller#Action
           recipes GET     /recipes(.:format)                   recipes#index
                      POST    /recipes(.:format)                   recipes#create
           new_recipe GET     /recipes/new(.:format)                 recipes#new
        edit_recipe GET     /recipes/:id/edit(.:format)             recipes#edit
          recipe GET     /recipes/:id(.:format)                 recipes#show
                     PATCH   /recipes/:id(.:format)                 recipes#update
                     PUT     /recipes/:id(.:format)                 recipes#update
                     DELETE  /recipes/:id(.:format)                 recipes#destroy
 rails_service_blob GET     /rails/active_storage/blobs/:signed_id/*filename(.:format) active_storage/blobs#show
rails_blob_representation GET   /rails/active_storage/representations/:signed_blob_id/:variation_key/*filename(.:format) active_storage/representations#show
 rails_disk_service GET   /rails/active_storage/disk/:encoded_key/*filename(.:format) active_storage/disk#show
 update_rails_disk_service PUT  /rails/active_storage/disk/:encoded_token(.:format) active_storage/disk#update
  rails_direct_uploads POST  /rails/active_storage/direct_uploads(.:format) active_storage/direct_uploads#create
```

Prefix é a rota nomeada.

CRIANDO O CONTROLLER

rails generate controller recipes

```
C:\Sites\receitasnet>rails generate controller recipes
      create  app/controllers/recipes_controller.rb
      invoke  erb
      create    app/views/recipes
      invoke  test_unit
      create    test/controllers/recipes_controller_test.rb
      invoke  helper
      create    app/helpers/recipes_helper.rb
      invoke  test_unit
      invoke  assets
      invoke    coffee
      create    app/assets/javascripts/recipes.coffee
      invoke    scss
      create    app/assets/stylesheets/recipes.scss
```

Além do controller, são criados outros arquivos auxiliares.

Por convenção o nome do controller deve estar sempre no **plural**.

app/controllers/recipes_controller.rb

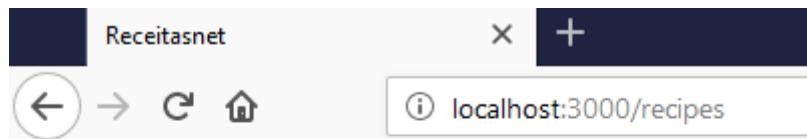
```
class RecipesController < ApplicationController
  def index
  end
end
```

CRIANDO A VIEW

app/views/recipes/index.html.erb

```
<h1>Receitas</h1>

<ul>
  <li>Strogonoff</li>
  <li>Filé de frango à milanesa</li>
  <li>Salmão ao forno</li>
</ul>
```



Receitas

- Strogonoff
- Filé de frango à milanesa
- Salmão ao forno

AULA 03 – ENTENDENDO V E C DO MVC – PARTE 2

Tornando a lista dinâmica

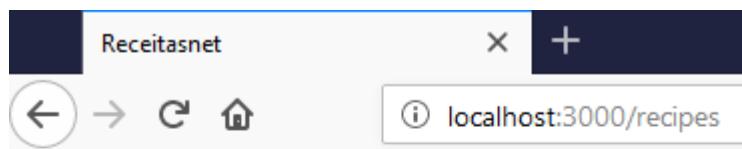
app/controllers/recipes_controller.rb

```
class RecipesController < ApplicationController
  def index
    @recipes = ['Strogonoff', 'Filé de frango à milanesa', 'Salmão ao forno', 'Bacalhau ao forno']
  end
end
```

app/views/recipes/index.html.erb

```
<h1>Receitas</h1>

<ul>
  <% @recipes.each do |recipe| %>
    <li><%= recipe %></li>
  <% end %>
</ul>
```



Receitas

- Strogonoff
- Filé de frango à milanesa
- Salmão ao forno
- Bacalhau ao forno

- Abra o arquivo Gemfile e faça as seguintes alterações:

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', '~> 5.2.1'
# 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 'ductape'
# 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 ActiveStorage 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
  # Use mysql as the database for Active Record
  gem 'mysql2', '>= 0.4.4', '< 0.6.0'
  # Access an interactive console on exception pages or by calling 'console' anywhere in the
  # code.
```

```
gem 'web-console', '>= 3.3.0'
# Call 'byebug' anywhere in the code to stop execution and get a debugger console
gem 'byebug', platforms: [:mri, :mingw, :x64_mingw]
end

group :test do
  # Adds support for Capybara system testing and selenium driver
  gem 'capybara', '>= 2.15'
  gem 'selenium-webdriver'
  # Easy installation and use of chromedriver to run system tests with Chrome
  gem 'chromedriver-helper'
end

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

bundle install --without production

```
C:\Sites\receitasnet>bundle install --without production
Fetching gem metadata from https://rubygems.org/...
Fetching gem metadata from https://rubygems.org/.
Resolving dependencies...
Using rake 13.0.0
Using concurrent-ruby 1.1.5
Using i18n 1.7.0
Using minitest 5.12.2
Using thread_safe 0.3.6
Using tzinfo 1.2.5
Using activesupport 5.2.3
Using builder 3.2.3
```

```
Using sass 3.7.4
Using tilt 2.0.10
Using sass-rails 5.1.0
Using selenium-webdriver 3.142.6
Using turbolinks-source 5.2.0
Using turbolinks 5.2.1
Using tzinfo-data 1.2019.3
Using uglifier 4.2.0
Using web-console 3.7.0
Bundle complete! 17 Gemfile dependencies, 76 gems now installed.
Gems in the group production were not installed.
Use `bundle info [gemname]` to see where a bundled gem is installed.
```

ALTERAÇÕES NOS ARQUIVOS DE CONFIGURAÇÃO

config/enviroments/production.rb

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

  # Code is not reloaded between requests.
  config.cache_classes = true

  # Eager load code on boot. This eager loads most of Rails and
  # your application in memory, allowing both threaded web servers
  # and those relying on copy on write to perform better.
  # Rake tasks automatically ignore this option for performance.
  config.eager_load = true

  # Full error reports are disabled and caching is turned on.
  config.consider_all_requests_local    = false
  config.action_controller.perform_caching = true

  # Ensures that a master key has been made available in either ENV["RAILS_MASTER_KEY"]
  # or in config/master.key. This key is used to decrypt credentials (and other encrypted
  # files).
  # config.require_master_key = true

  # Disable serving static files from the `/public` folder by default since
  # Apache or NGINX already handles this.
  #config.public_file_server.enabled = ENV['RAILS_SERVE_STATIC_FILES'].present?
  config.public_file_server.enabled = true

  # Compress JavaScripts and CSS.
  config.assets.js_compressor = :uglifier
  # config.assets.css_compressor = :sass

  # Do not fallback to assets pipeline if a precompiled asset is missed.
  config.assets.compile = false

  # `config.assets.precompile` and `config.assets.version` have moved to
  config/initializers/assets.rb

  # Enable serving of images, stylesheets, and JavaScripts from an asset server.
  # config.action_controller.asset_host = 'http://assets.example.com'
```

```
# Specifies the header that your server uses for sending files.
# config.action_dispatch.x_sendfile_header = 'X-Sendfile' # for Apache
# config.action_dispatch.x_sendfile_header = 'X-Accel-Redirect' # for NGINX

# Store uploaded files on the local file system (see config/storage.yml for options)
config.active_storage.service = :local

# Mount Action Cable outside main process or domain
# config.action_cable.mount_path = nil
# config.action_cable.url = 'wss://example.com/cable'
# config.action_cable.allowed_request_origins = [ 'http://example.com',
# http://\Vexample.*' ]

# Force all access to the app over SSL, use Strict-Transport-Security, and use secure
cookies.
# config.force_ssl = true

# Use the lowest log level to ensure availability of diagnostic information
# when problems arise.
config.log_level = :debug

# Prepend all log lines with the following tags.
config.log_tags = [ :request_id ]

# Use a different cache store in production.
# config.cache_store = :mem_cache_store

# Use a real queuing backend for Active Job (and separate queues per environment)
# config.active_job.queue_adapter = :resque
# config.active_job.queue_name_prefix = "receitasnet_#{Rails.env}"

config.action_mailer.perform_caching = false

# Ignore bad email addresses and do not raise email delivery errors.
# Set this to true and configure the email server for immediate delivery to raise delivery
errors.
# config.action_mailer.raise_delivery_errors = false

# Enable locale fallbacks for I18n (makes lookups for any locale fall back to
# the I18n.default_locale when a translation cannot be found).
config.i18n.fallbacks = true

# Send deprecation notices to registered listeners.
config.active_support.deprecation = :notify
```

```

# Use default logging formatter so that PID and timestamp are not suppressed.
config.log_formatter = ::Logger::Formatter.new

# Use a different logger for distributed setups.
# require 'syslog/logger'
# config.logger = ActiveSupport::TaggedLogging.new(Syslog::Logger.new 'app-name')

if ENV["RAILS_LOG_TO_STDOUT"].present?
  logger      = ActiveSupport::Logger.new(STDOUT)
  logger.formatter = config.log_formatter
  config.logger = ActiveSupport::TaggedLogging.new(logger)
end

# Do not dump schema after migrations.
config.active_record.dump_schema_after_migration = false
end

```

CONTROLADOR DE VERSÃO - GIT

Preparando o ambiente no terminal

```

git config --global user.name "Roberto Pinheiro"
git config --global user.email "betopinheiro1005@yahoo.com.br"

```

```

git init
git status
git add .
git status
git commit -m "Commit inicial"

```

- Abrir uma conta no Heroku

- Baixar e instalar heroku cli (ou heroku toolbet) – Controle por linha de comando

```
heroku login
```

```

C:\Sites\receitasnet>heroku login
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.herokuapp.com/auth/browser/33d45844-e9b4-4c5a-b593-7a10c26b91cc
Logging in... done
Logged in as beto10561@gmail.com

```

Gerando chave de segurança

heroku keys:add

```
C:\Sites\receitasnet>heroku keys:add
Found an SSH public key at C:\Users\pinhe\.ssh\id_rsa.pub
? Would you like to upload it to Heroku? Yes
Uploading C:\Users\pinhe\.ssh\id_rsa.pub SSH key... done
```

Criando a aplicação

heroku apps:create receitasnet-robertopinheiro

```
C:\Sites\receitasnet>heroku apps:create receitasnet-robertopinheiro
Creating ⬤ receitasnet-robertopinheiro... done
https://receitasnet-robertopinheiro.herokuapp.com/ | https://git.heroku.com/receitasnet-robertopinheiro.git
```

git remote -v

```
C:\Sites\receitasnet>git remote -v
heroku  https://git.heroku.com/receitasnet-robertopinheiro.git (fetch)
heroku  https://git.heroku.com/receitasnet-robertopinheiro.git (push)
```

Fazendo o deploy

git push heroku master

Informando ao Heroku qual é a página inicial do aplicativo

config/routes.rb

```
Rails.application.routes.draw do
  root 'recipes#index'
  resources :recipes
end
```

Criando o banco de dados no heroku

heroku run rake db:migrate

```
C:\Sites\receitasnet>heroku run rake db:migrate
Running `rake db:migrate` on ⬤ receitasnet-robetopinheiro... up, run.4439 (Free)
D, [2019-10-08T21:07:17.493405 #4] DEBUG -- :   (15.0ms) CREATE TABLE "schema_migrations" ("version" character varying NOT NULL PRIMARY KEY)
D, [2019-10-08T21:07:17.505561 #4] DEBUG -- :   (8.5ms) CREATE TABLE "ar_internal_metadata" ("key" character varying NOT NULL PRIMARY KEY, "value" character varying, "created_at" timestamp NOT NULL, "updated_at" timestamp NOT NULL)
D, [2019-10-08T21:07:17.509739 #4] DEBUG -- :   (1.9ms) SELECT pg_try_advisory_lock(5279828704805601030)
D, [2019-10-08T21:07:17.533830 #4] DEBUG -- :   (1.9ms) SELECT "schema_migrations"."version" FROM "schema_migrations" ORDER BY "schema_migrations"."version" ASC
D, [2019-10-08T21:07:17.559922 #4] DEBUG -- : ActiveRecord::InternalMetadata Load (2.1ms) SELECT "ar_internal_metadata".* FROM "ar_internal_metadata" WHERE "ar_internal_metadata"."key" = $1 LIMIT $2 [["key", "environment"], ["LIMIT", 1]]
D, [2019-10-08T21:07:17.579006 #4] DEBUG -- :   (1.1ms) BEGIN
D, [2019-10-08T21:07:17.583038 #4] DEBUG -- : ActiveRecord::InternalMetadata Create (1.4ms) INSERT INTO "ar_internal_metadata" ("key", "value", "created_at", "updated_at") VALUES ($1, $2, $3, $4) RETURNING "key" [["key", "environment"], ["value", "production"], ["created_at", "2019-10-08 21:07:17.580596"], ["updated_at", "2019-10-08 21:07:17.580596"]]
D, [2019-10-08T21:07:17.590029 #4] DEBUG -- :   (6.6ms) COMMIT
D, [2019-10-08T21:07:17.592264 #4] DEBUG -- :   (1.6ms) SELECT pg_advisory_unlock(5279828704805601030)
```

Atualizando o repositório

git status

```
C:\Sites\receitasnet>git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
    (use "git restore <file>..." to discard changes in working directory)
      modified:   config/routes.rb

no changes added to commit (use "git add" and/or "git commit -a")
```

git add config/routes.rb

```
C:\Sites\receitasnet>git add config/routes.rb
warning: LF will be replaced by CRLF in config/routes.rb.
The file will have its original line endings in your working directory
```

git commit -m "Inserindo a rota raiz"

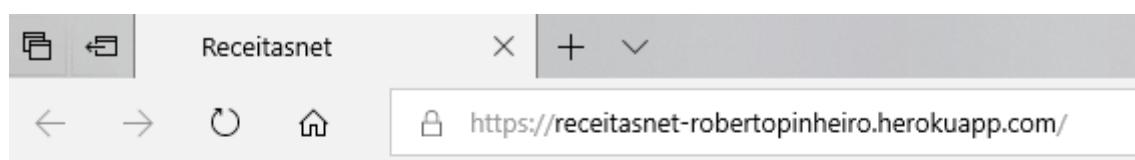
```
C:\Sites\receitasnet>git commit -m "Inserindo a rota raiz"
[master 8d4d6f1] Inserindo a rota raiz
 1 file changed, 1 insertion(+)
```

```
git push heroku master
```

Abrindo a aplicação

```
heroku open
```

<https://receitasnet-robertopinheiro.herokuapp.com/>



Receitas

- Strogonoff
- Filé de frango à milanesa
- Salmão ao forno
- Bacalhau ao forno

AULA 04 – MODELOS

A principal função do modelo é armazenar as informações do projeto e guardar as regras de negócio da aplicação.

Modelos

```
class Recipe < ActiveRecord::Base  
end
```

id	name	stuff	calories
1	Estrogonofe de carne	1 kg de carne,3 cebolas médias ...	100
2	Filé de frango a parmegiana	1/2 kg de filé de frango ,2 ovos batidos...	160
2	Salmão ao forno	1 salmão,Sal grosso,Azeite	110

O nome da tabela, por padrão, vem no **plural**.

CRIANDO O MODEL

No terminal, entre com o comando:

```
rails generate model Recipe name:string stuff:text calories:integer
```

O nome do modelo é sempre no **singular**. E a **primeira letra maiúscula**.

```
C:\Sites\receitasnet>rails generate model Recipe name stuff:text calories:integer  
      invoke  active_record  
      create    db/migrate/20191008220014_create_recipes.rb  
      create    app/models/recipe.rb  
      invoke  test_unit  
      create    test/models/recipe_test.rb  
      create    test/fixtures/recipes.yml
```

Além do model, é criado uma migration e alguns outros arquivos.

db/migrate/ 20191008220014_create_recipes.rb

```
class CreateRecipes < ActiveRecord::Migration[5.2]
  def change
    create_table :recipes do |t|
      t.string :name
      t.text :stuff
      t.integer :calories

      t.timestamps
    end
  end
end
```

CRIANDO A TABELA RECIPES

rake db:migrate

```
C:\Sites\receitasnet>rake db:migrate
== 20191008220014 CreateRecipes: migrating ==
-- create_table(:recipes)
 -> 0.3452s
== 20191008220014 CreateRecipes: migrated (0.3460s) ==
```

▼	receitasnet_development	48,0 KiB
	ar_internal_metadata	16,0 KiB
★	recipes	16,0 KiB
	schema_migrations	16,0 KiB
➤	receitasnet_test	

Colunas:	Adicionar	Remover	Mover para cima	Mover para baixo
#	Nome	Tipo de dados	Tamanho/Itens	Unsign... Permitir... Zerofill Padrão
1	id	BIGINT	20	<input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> AUTO_INCREMENT
2	name	VARCHAR	255	<input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> NULL
3	stuff	TEXT		<input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> NULL
4	calories	INT	11	<input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> NULL
5	created_at	DATETIME		<input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> Nenhum padrão
6	updated_at	DATETIME		<input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> Nenhum padrão

Tarefas que podem ser executadas com o rake

rake -T (ou rake --tasks)

```
C:\Sites\receitasnet>rake -T
rake about
rake active_storage:install
rake app:template
rake app:update
rake assets:clean[keep]
rake assets:clobber
rake assets:environment
rake assets:precompile
rake cache_digests:dependencies
rake cache_digests:nested_dependencies
rake db:create
rake db:drop
rake db:environment:set
rake db:fixtures:load
rake db:migrate
rake db:migrate:status
rake db:rollback
rake db:schema:cache:clear
rake db:schema:cache:dump
rake db:schema:dump
rake db:schema:load
rake db:seed
rake db:setup
rake db:structure:dump
rake db:structure:load
rake db:version

rake dev:cache
rake initializers
rake log:clear
rake middleware
rake notes
rake notes:custom
rake restart
rake routes
rake secret
rake stats
rake test
rake test:db
rake test:system
rake time:zones[country_or_offset]
rake tmp:clear
rake tmp:create
rake yarn:install

# List versions of all Rails framework...
# Copy over the migration needed to...
# Applies the template supplied by ...
# Update configs and some other ini...
# Remove old compiled assets
# Remove compiled assets
# Load asset compile environment
# Compile all the assets named in config/...
# Lookup first-level dependencies for ...
# Lookup nested dependencies for TestEnvi...
# Creates the database from DATABASE...
# Drops the database from DATABASE...
# Set the environment value for the current...
# Loads fixtures into the current environment
# Migrate the database (options: VERSION...
# Display status of migrations
# Rolls the schema back to the previous ...
# Clears a db/schema_cache.yml file
# Creates a db/schema_cache.yml file
# Creates a db/schema.rb file that ...
# Loads a schema.rb file into the database
# Loads the seed data from db/seeds.rb
# Creates the database, loads the schema ...
# Dumps the database structure to disk
# Recreates the databases from the schema ...
# Retrieves the current schema version
# Toggle development mode caching options
# Print out all defined initializers
# Truncates all/specify *.log files
# Prints out your Rack middleware stack
# Enumerate all annotations (use no ...
# Enumerate a custom annotation, specifying ...
# Restart app by touching tmp/restart ...
# Print out all defined routes in memory
# Generate a cryptographically secure ...
# Report code statistics (KLOCs, etc)
# Runs all tests in test folder excluding ...
# Run tests quickly, but also reset db ...
# Run system tests only
# List all time zones, list by two-letter ...
# Clear cache, socket and screenshots
# Creates tmp directories for cache ...
# Install all JavaScript dependencies
```

Status da migração

rake db:migrate:status

```
C:\Sites\receitasnet>rake db:migrate:status

database: receitasnet_development

  Status  Migration ID  Migration Name
  -----
    up      20191008220014  Create recipes
```

INSERINDO DADOS PELO TERMINAL

Console do Rails

`rails c`

Primeira forma:

Recipe

```
r = Recipe.new
r.name = 'Strogonoff de carne'
r.stuff = ' 1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup'
r.calories = 100
```

`r.save`

```
=> Recipe (call 'Recipe.connection' to establish a connection)
irb(main):002:0> r = Recipe.new
  (0.6ms)  SET NAMES utf8, @SESSION.sql_mode = CONCAT(CONCAT(@@sql_mode, ',STRICT_ALL_TABLES'), ',NO_AUTO_VALUE_ON_ZERO'), @SESSION.sql_auto_is_null = 0, @SESSION.wait_timeout = 2147483
=> #<Recipe id: nil, name: nil, stuff: nil, calories: nil, created_at: nil, updated_at: nil>
irb(main):003:0> r.name = 'Strogonoff de carne'
=> "Strogonoff de carne"
irb(main):004:0> r.stuff = ' 1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup'
=> " 1 kg de carne, 3 cebolas m\x82dias, sal e pimenta-do-reino \x85 gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup"
irb(main):005:0> r.calories = 100
=> 100
irb(main):006:0> r.save
  (0.5ms)  BEGIN
    Recipe Create (67.8ms)  INSERT INTO `recipes` (`name`, `stuff`, `calories`, `created_at`, `updated_at`)
VALUES ('Strogonoff de carne', ' 1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup', 100, '2019-10-09 00:45:45', '2019-10-09 00:45:45')
    (52.6ms)  COMMIT
=> true
```

receitasnet_development.recipes: 1 registros totais (aproximadamente)						Próximo	Mostrar todos	Ordem	Colunas (6/6)
#	id	name	stuff	calories	created_at	updated_at			
	1	Strogonoff de carne	1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup	100	2019-10-09 00:45:45	2019-10-09 00:45:45			

Segunda forma:

Recipe

```
r = Recipe.new(name:'Filé de frango à parmegiana', stuff:'1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 xícaras de chá de farinha de rosca, 1 caixinha de molho de tomate pronto, mussarela', calories: 160)
```

`r.save`

```

irb(main):001:0> Recipe
=> Recipe (call 'Recipe.connection' to establish a connection)
<1, 2 xícaras de chá de farinha de rosca, 1 caixinha de molho de tomate pronto, mussarela', calories: 160)
  (0.7ms)  SET NAMES utf8, @@SESSION.sql_mode = CONCAT(CONCAT(@@sql_mode, ',STRICT_ALL_TABLES'), ',NO_AUTO_VALUE_ON_ZE
R0', @@SESSION.sql_auto_is_null = 0, @@SESSION.wait_timeout = 2147483
=> #<Recipe id: nil, name: "Filé\x82 de frango", stuff: "1/2 kg de filé\x82 de frango, 2 ovos batidos com 1 pit...", calor
ies: 160, created_at: nil, updated_at: nil>
irb(main):003:0> r.save
  (0.4ms)
  Recipe Create (66.5ms)  INSERT INTO `recipes` (`name`, `stuff`, `calories`, `created_at`, `updated_at`) VALUES ('Filé
de frango', '1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 xícaras de chá de farinha de rosca, 1 caixi
nha de molho de tomate pronto, mussarela', 160, '2019-10-09 01:23:38', '2019-10-09 01:23:38')
  (91.0ms)  COMMIT
=> true

```

receitasnet_development.recipes: 3 registros totais (aproximadamente)					
	id	name	stuff	calories	created_at
	1	Strogonoff de carne	1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino ...	100	2019-10-11 23:42:27
	2	Filé de frango à parmegiana	1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de ...	160	2019-10-11 23:43:00
	3	Salmão ao forno	1 salmão, sal grosso e azeite	110	2019-10-11 23:43:20

Terceira forma:

```
r = Recipe.create(name:'Salmão ao forno', stuff:'1 salmão, sal grosso e azeite ', calories: 110)
```

```

irb(main):004:0> r = Recipe.create(name:'Salmão ao forno', stuff:'1 salmão, sal grosso e azeite ', calories: 110)
  (0.6ms)  BEGIN
  Recipe Create (96.4ms)  INSERT INTO `recipes` (`name`, `stuff`, `calories`, `created_at`, `updated_at`) VALUES ('Salmâ
o ao forno', '1 salmão, sal grosso e azeite ', 110, '2019-10-09 01:32:06', '2019-10-09 01:32:06')
  (105.3ms)  COMMIT
=> #<Recipe id: 3, name: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite ", calories: 110, created_at: "2
019-10-09 01:32:06", updated_at: "2019-10-09 01:32:06">
irb(main):005:0>

```

receitasnet_development.recipes: 3 registros totais (aproximadamente)					
	id	name	stuff	calories	created_at
	1	Strogonoff de carne	1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite...	100	2019-10-09 00:45:45
	2	Filé de frango	1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 xícaras de chá de farinha de...	160	2019-10-09 01:23:38
	3	Salmão ao forno	1 salmão, sal grosso e azeite	110	2019-10-09 01:32:06

Listando

Recipe.all

```

irb(main):005:0> Recipe.all
  Recipe Load (0.9ms)  SELECT `recipes`.* FROM `recipes` LIMIT 11
=> #<ActiveRecord::Relation [#<Recipe id: 1, name: "Strogonoff de carne", stuff: " 1 kg de carne, 3 cebolas médias, sal
e pimenta-do...", calories: 100, created_at: "2019-10-09 00:45:45", updated_at: "2019-10-09 00:45:45">, #<Recipe id: 2,
name: "Filé de frango", stuff: "1/2 kg de filé de frango, 2 ovos batidos com 1 pit...", calories: 160, created_at: "2019
-10-09 01:23:38", updated_at: "2019-10-09 01:23:38">, #<Recipe id: 3, name: "Salmão ao forno", stuff: "1 salmão, sal gro
sso e azeite ", calories: 110, created_at: "2019-10-09 01:32:06", updated_at: "2019-10-09 01:32:06">]

```

Último registro

Recipe.last

```

irb(main):009:0> Recipe.last
  Recipe Load (0.9ms)  SELECT `recipes`.* FROM `recipes` ORDER BY `recipes`.`id` DESC LIMIT 1
=> #<Recipe id: 3, name: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite", calories: 110, created_at: "2019-10
-08 23:15:05", updated_at: "2019-10-08 23:15:05">

```

Primeiro registro

Recipe.first

```
irb(main):010:0> Recipe.first
  Recipe Load (0.9ms)  SELECT `recipes`.* FROM `recipes` ORDER BY `recipes`.`id` ASC LIMIT 1
=> #<Recipe id: 1, name: "Strogonoff de carne", stuff: "1 kg de carne, 3 cebolas médias, sal e pimenta-do...", calories: 100, created_at: "2019-10-08 22:50:47", updated_at: "2019-10-08 22:50:47">
```

Quantidade de registros

Recipe.count

```
irb(main):011:0> Recipe.count
  (30.3ms)  SELECT COUNT(*) FROM `recipes`
=> 3
```

Alterando um registro

Inicialmente é necessário selecionar o registro a ser atualizado.

receita = Recipe.find(1)

```
irb(main):012:0> receita = Recipe.find(1)
  Recipe Load (62.7ms)  SELECT `recipes`.* FROM `recipes` WHERE `recipes`.`id` = 1 LIMIT 1
=> #<Recipe id: 1, name: "Strogonoff de carne", stuff: "1 kg de carne, 3 cebolas médias, sal e pimenta-do...", calories: 100, created_at: "2019-10-08 22:50:47", updated_at: "2019-10-08 22:50:47">
```

Primeira forma:

```
receita = Recipe.find(1)
receita.name = 'Bacalhau ao forno'
receita.stuff = '1 posta de bacalhau (parte central ou filé, 4 dentes de alho, 1 cebola grande, e azeitonas pretas com caroço)'
receita.calories = 250
receita.save
```

```
irb(main):006:0> receita = Recipe.find(1)
  Recipe Load (0.9ms)  SELECT `recipes`.* FROM `recipes` WHERE `recipes`.`id` = 1 LIMIT 1
=> #<Recipe id: 1, name: "Strogonoff de carne", stuff: "1 kg de carne, 3 cebolas médias, sal e pimenta-do...", calories: 100, created_at: "2019-10-09 00:45:45", updated_at: "2019-10-09 00:45:45">
irb(main):007:0> receita.name = 'Bacalhau ao forno'
=> "Bacalhau ao forno"
< de bacalhau (parte central ou filé, 4 dentes de alho, 1 cebola grande, e azeitonas pretas com caroço)'
=> "1 posta de bacalhau (parte central ou filé, 4 dentes de alho, 1 cebola grande, e azeitonas pretas com caroço)"

irb(main):008:0> receita.calories = 250
=> 250
irb(main):010:0> receita.save
  (0.6ms)  BEGIN
  Recipe Update (1.0ms)  UPDATE `recipes` SET `name` = 'Bacalhau ao forno', `stuff` = '1 posta de bacalhau (parte central ou filé, 4 dentes de alho, 1 cebola grande, e azeitonas pretas com caroço)', `calories` = 250, `updated_at` = '2019-10-09 01:42:53' WHERE `recipes`.`id` = 1
  (116.3ms)  COMMIT
=> true
```

receitasnet_development.recipes: 3 registros totais (aproximadamente)			
		Próximo	Mostrar todos Ordem Colunas (6/6)
ação	id	name	stuff
	1	Bacalhau ao forno	1 posta de bacalhau (parte central ou filé, 4 dentes de alho, 1 cebola grande, e azeitonas ...
	2	Filé de frango	1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 xícaras de chá de farinha de...
	3	Salmão ao forno	1 salmão, sal grosso e azeite

Segunda forma:

```
receita = Recipe.find(1)
```

```
receita.update(name:'Strogonoff de carne', stuff:'1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup', calories: 100 )
```

```
irb(main):011:0> receita = Recipe.find(1)
  Recipe Load (1.2ms)  SELECT `recipes`.* FROM `recipes` WHERE `recipes`.`id` = 1 LIMIT 1
=> #<Recipe id: 1, name: "Bacalhau ao forno", stuff: "1 posta de bacalhau (parte central ou filé, 4 dent...", calories: 250, created_at: "2019-10-09 00:45:45", updated_at: "2019-10-09 01:42:53">
<o à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup", calories: 100 )
  (0.4ms)  BEGIN
  Recipe Update (18.2ms)  UPDATE `recipes` SET `name` = 'Strogonoff de carne', `stuff` = '1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup', `calories` = 100, `updated_at` = '2019-10-09 01:50:51' WHERE `recipes`.`id` = 1
  (61.6ms)  COMMIT
=> true
```

receitasnet_development.recipes: 3 registros totais (aproximadamente)			
		Próximo	Mostrar todos Ordem Colunas (6/6)
ação	id	name	stuff
	1	Strogonoff de carne	1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, ...
	2	Filé de frango	1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 xícaras de chá de farinha de...
	3	Salmão ao forno	1 salmão, sal grosso e azeite

Deletando um registro

Primeiro é necessário selecionar o registro a ser excluído.

```
r = Recipe.find_by(name:"Salmão ao forno")
r.destroy
```

```
irb(main):015:0> r = Recipe.find_by(name:"Salmão ao forno")
  Recipe Load (0.4ms)  SELECT `recipes`.* FROM `recipes` WHERE `recipes`.`name` = 'Salmão ao forno' LIMIT 1
=> #<Recipe id: 3, name: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite ", calories: 110, created_at: "2019-10-09 01:32:06", updated_at: "2019-10-09 01:32:06">
irb(main):016:0> r.destroy
  (0.4ms)  BEGIN
  Recipe Destroy (0.8ms)  DELETE FROM `recipes` WHERE `recipes`.`id` = 3
  (256.9ms)  COMMIT
=> #<Recipe id: 3, name: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite ", calories: 110, created_at: "2019-10-09 01:32:06", updated_at: "2019-10-09 01:32:06">
```

receitasnet_development.recipes: 2 registros totais (aproximadamente)			
		Próximo	Mostrar todos Ordem Colunas (6/6)
ação	id	name	stuff
	1	Strogonoff de carne	1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, ...
	2	Filé de frango	1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 xícaras de chá de farinha de...

ADICIONANDO AS ALTERAÇÕES NO GIT

git add .

```
C:\Sites\receitasnet>git add .
warning: LF will be replaced by CRLF in Gemfile.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in app/models/recipe.rb.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in db/migrate/20191008220014_create_recipes.rb.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in test/fixtures/recipes.yml.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in test/models/recipe_test.rb.
The file will have its original line endings in your working directory
```

git status

```
C:\Sites\receitasnet>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   Gemfile
    new file:   app/models/recipe.rb
    new file:   db/migrate/20191008220014_create_recipes.rb
    modified:   db/schema.rb
    new file:   test/fixtures/recipes.yml
    new file:   test/models/recipe_test.rb
```

git commit -m "Criando model Recipe"

```
C:\Sites\receitasnet>git commit -m "Criando model Recipe"
[master aa23460] Criando model Recipe
 6 files changed, 48 insertions(+), 4 deletions(-)
 create mode 100644 app/models/recipe.rb
 create mode 100644 db/migrate/20191008220014_create_recipes.rb
 create mode 100644 test/fixtures/recipes.yml
 create mode 100644 test/models/recipe_test.rb
```

git push heroku master

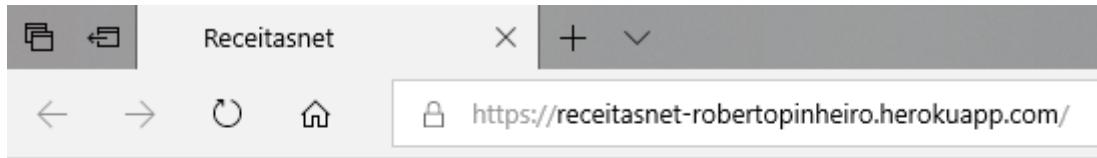
Rodando a migration no heroku

heroku run rake db:migrate

```
C:\Sites\receitasnet>heroku run rake db:migrate
Running rake db:migrate on ● receitasnet-robetopinheiro... up, run.5272 (Free)
D, [2019-10-09T02:11:34.309745 #4] DEBUG -- :   (0.7ms)  SELECT pg_try_advisory_lock(5279828704805601030)
D, [2019-10-09T02:11:34.326463 #4] DEBUG -- :   (1.6ms)  SELECT "schema_migrations"."version" FROM "schema_migrations" ORDER BY "schema_migrations"."version" ASC
I, [2019-10-09T02:11:34.330053 #4] INFO -- : Migrating to CreateRecipes (20191008220014)
D, [2019-10-09T02:11:34.334036 #4] DEBUG -- :   (0.6ms)  BEGIN
== 20191008220014 CreateRecipes: migrating =====
-- create_table(recipes)
D, [2019-10-09T02:11:34.349694 #4] DEBUG -- :   (13.8ms)  CREATE TABLE "recipes" ("id" bigserial primary key, "name" character varying, "stuff" text, "calories" integer, "created_at" timestamp NOT NULL, "updated_at" timestamp NOT NULL)
   -> 0.0150s
== 20191008220014 CreateRecipes: migrated (0.0153s) =====

D, [2019-10-09T02:11:34.357987 #4] DEBUG -- : ActiveRecord::SchemaMigration Create (0.8ms)  INSERT INTO "schema_migrations" ("version") VALUES ($1) RETURNING "version" [[{"version": "20191008220014"}]]
D, [2019-10-09T02:11:34.360119 #4] DEBUG -- :   (1.8ms)  COMMIT
D, [2019-10-09T02:11:34.366578 #4] DEBUG -- : ActiveRecord::InternalMetadata Load (0.8ms)  SELECT "ar_internal_metadata",* FROM "ar_internal_metadata" WHERE "ar_internal_metadata"."key" = $1 LIMIT $2 [[{"key": "environment"}, {"LIMIT": 1}]]
D, [2019-10-09T02:11:34.375390 #4] DEBUG -- :   (0.5ms)  BEGIN
D, [2019-10-09T02:11:34.376859 #4] DEBUG -- :   (0.5ms)  COMMIT
D, [2019-10-09T02:11:34.377829 #4] DEBUG -- :   (0.7ms)  SELECT pg_advisory_unlock(5279828704805601030)
```

Heroku open

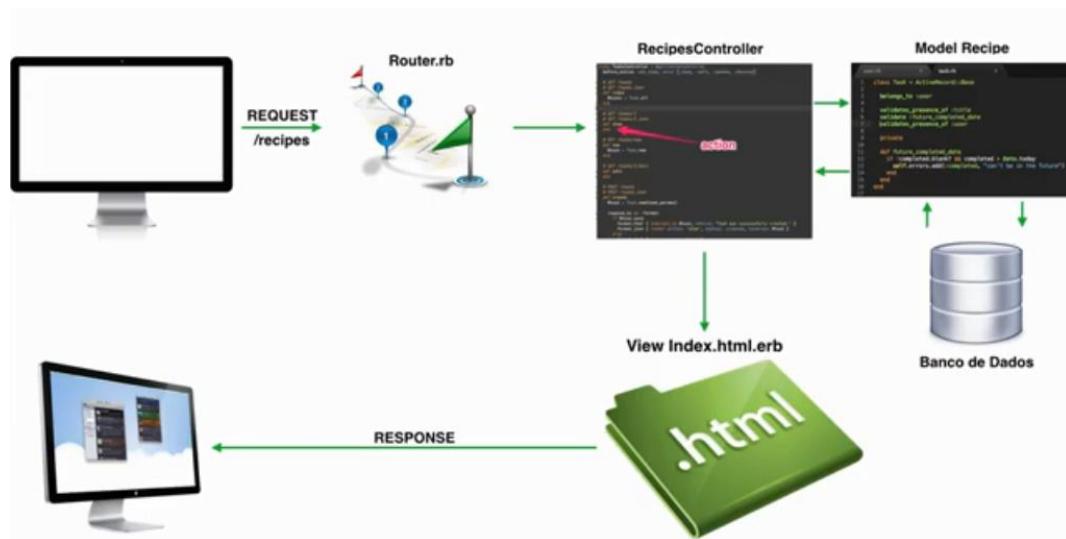


Receitas

- Strogonoff
- Filé de frango à milanesa
- Salmão ao forno
- Bacalhau ao forno

AULA 05 – INTEGRANDO TUDO

PADRÃO MVC



`app/controllers/recipes_controller.rb`

```
class RecipesController < ApplicationController
  def index
    @recipes = Recipe.all
  end
end
```

`app/views/recipes/index.html.erb`

```
<h1>Receitas</h1>
<ul>
  <% @recipes.each do |recipe| %>
    <li>Nome: <%= recipe.name %></li>
    <li>Ingredientes: <%= recipe.stuff %></li>
    <li>Calorias: <%= recipe.calories %></li>
    <hr>
  <% end %>
</ul>
```



Receitas

- Nome: Strogonoff de carne
 - Ingredientes: 1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup
 - Calorias: 100
-
- Nome: Filé de frango
 - Ingredientes: 1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 xícaras de chá de farinha de rosca, 1 caixinha de molho de tomate pronto, mussarela
 - Calorias: 160

Inserindo uma nova receita pela linha de comando

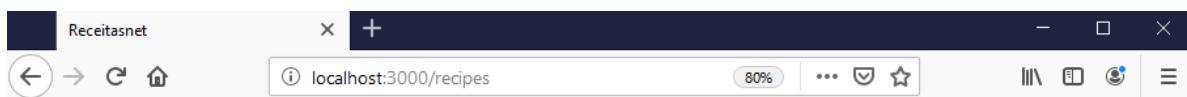
rails c

Recipe

```
r = Recipe.create(name:'Salmão ao forno', stuff:'1 salmão, sal grosso e azeite ', calories: 110)
```

```
C:\Sites\receitasnet>rails c
Loading development environment (Rails 5.2.3)
irb(main):001:0> Recipe
=> Recipe (call 'Recipe.connection' to establish a connection)
irb(main):002:0> r = Recipe.create(name:'Salmão ao forno', stuff:'1 salmão, sal grosso e azeite ', calories: 110)
               (0.5ms)  SET NAMES utf8, @@SESSION.sql_mode = CONCAT(CONCAT(@@sql_mode, ',STRICT_ALL_TABLES'), ',NO_AUTO_VALUE_ON_ZERO'), @@SESSION.sql_auto_is_null = 0, @@SESSION.wait_timeout = 21
47483
               (0.3ms)  BEGIN
               Recipe Create (16.1ms)  INSERT INTO `recipes` (`name`, `stuff`, `calories`, `created_at`, `updated_at`) VALUES ('Salmão ao forno', '1 salmão, sal grosso e azeite ', 110, '2019-10-09 02:34:04', '2019-10-09 02:34:04')
               (54.1ms)  COMMIT
=> #<Recipe id: 4, name: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite ", calories: 110, created_at: "2019-10-09 02:34:04", updated_at: "2019-10-09 02:34:04">
```

receitasnet_development.recipes: 3 registros totais (aproximadamente)					Próximo	Mostrar todos	Ordem	Colunas (6/6)
	id	name	stuff	calories	created_at	updated_at		
	1	Strogonoff de carne	1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup	100	2019-10-09 00:45:45	2019-10-09 01:50:51		
	2	Filé de frango	1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 xícaras de chá de farinha de rosca, 1 caixinha de molho de tomate pronto, mussarela	160	2019-10-09 01:23:38	2019-10-09 01:23:38		
	4	Salmão ao forno	1 salmão, sal grosso e azeite	110	2019-10-09 02:34:04	2019-10-09 02:34:04		



Receitas

- Nome: Strogonoff de carne
 - Ingredientes: 1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup
 - Calorias: 100
-
- Nome: Filé de frango
 - Ingredientes: 1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 xícaras de chá de farinha de rosca, 1 caixinha de molho de tomate pronto, mussarela
 - Calorias: 160
-
- Nome: Salmão ao forno
 - Ingredientes: 1 salmão, sal grosso e azeite
 - Calorias: 110

ADICIONANDO AS ALTERAÇÕES NO GIT

git status

```
C:\Sites\receitasnet>git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   app/controllers/recipes_controller.rb
    modified:   app/views/recipes/index.html.erb

no changes added to commit (use "git add" and/or "git commit -a")
```

git add .

```
C:\Sites\receitasnet>git add .
warning: LF will be replaced by CRLF in app/controllers/recipes_controller.rb.
The file will have its original line endings in your working directory
```

git status

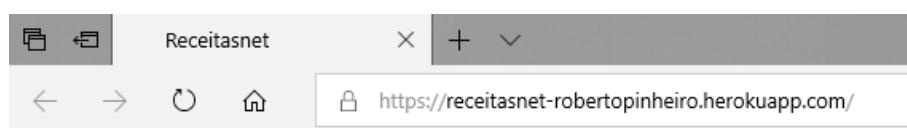
```
C:\Sites\receitasnet>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   app/controllers/recipes_controller.rb
    modified:   app/views/recipes/index.html.erb
```

git commit -m "Retira hardcoded do controller e ajusta a view"

```
C:\Sites\receitasnet>git commit -m "Retira hardcoded do controller e ajusta a view"
[master 0555b13] Retira hardcoded do controller e ajusta a view
 2 files changed, 5 insertions(+), 2 deletions(-)
```

git push heroku master

heroku open



Receitas

```
C:\Sites\receitasnet>git log
commit 388ff89c6b1326946a1f594856507a574e006aff (HEAD -> master, heroku/master)
Author: Roberto Pinheiro <betopinheiro1005@yahoo.com.br>
Date:   Fri Oct 11 20:52:33 2019 -0300

    Retira hardcoded do controller e ajusta a view

commit 41ed22ee0a60d8ba1bebbacad2c5363e717c7215
Author: Roberto Pinheiro <betopinheiro1005@yahoo.com.br>
Date:   Fri Oct 11 20:45:51 2019 -0300

    Criando model Recipe

commit 354919f6cf0a1f071b95e6add80da654e6380dc0
Author: Roberto Pinheiro <betopinheiro1005@yahoo.com.br>
Date:   Fri Oct 11 20:34:16 2019 -0300

    Inserindo a rota raiz

commit a11b48f30ca770cbc3b409ab6268102aa3c19c08
Author: Roberto Pinheiro <betopinheiro1005@yahoo.com.br>
Date:   Fri Oct 11 20:17:44 2019 -0300

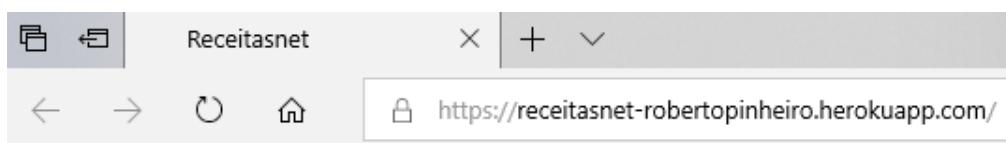
    Commit inicial
```

heroku run console

```
C:\Sites\receitasnet>heroku run console
Running console on ⚡ receitasnet-robertopinheiro... up, run.9827 (Free)
Loading production environment (Rails 5.2.3)
irb(main):001:0> |
```

r = Recipe.create(name:'Salmão ao forno', stuff:'1 salmão, sal grosso e azeite ', calories: 110)

```
irb(main):015:0> r = Recipe.create(name:'Salmão ao forno', stuff:'1 salmão, sal grosso e azeite', calories: 110)
D, [2019-10-09T02:54:53.945932 #4] DEBUG -- :   (0.8ms)  BEGIN
D, [2019-10-09T02:54:53.948594 #4] DEBUG -- :   Recipe Create (1.2ms)  INSERT INTO "recipes" ("name", "stuff", "calories", "created_at", "updated_at") VALUES ($1, $2, $3, $4, $5) RETURNING "id"  [["name", "Salmão ao forno"], ["stuff", "1 salmão, sal grosso e azeite"], ["calories", 110], ["created_at", "2019-10-09 02:54:53.946228"], ["updated_at", "2019-10-09:54:53.946228"]]
D, [2019-10-09T02:54:53.950434 #4] DEBUG -- :   (1.5ms)  COMMIT
=> #<Recipe id: 1, name: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite", calories: 110, created_at: "2019-10-09 02:54:53", updated_at: "2019-10-09 02:54:53">
```



Receitas

- Nome: Salmão ao forno
- Ingredientes: 1 salmão, sal grosso e azeite
- Calorias: 110

AULA 06 – MIGRAÇÕES

Criando uma migration para adicionar campos na tabela recipes

```
rails g migration AddNewFieldsToRecipes prepare_mode:text cost:decimal
```

```
C:\Sites\receitasnet>rails g migration AddNewFieldsToRecipes prepare_mode:text cost:decimal
      invoke  active_record
      create    db/migrate/20191009145025_add_new_fields_to_recipes.rb
```

db/migrate/20191009145025_add_new_fields_to_recipes.rb

```
class AddNewFieldsToRecipes < ActiveRecord::Migration[5.2]
  def change
    add_column :recipes, :prepare_mode, :text
    add_column :recipes, :cost, :decimal
  end
end
```

rake db:migrate:status

```
C:\Sites\receitasnet>rake db:migrate:status

database: receitasnet_development

  Status  Migration ID    Migration Name
  -----
    up      20191008220014  Create recipes
  down     20191009145025  Add new fields to recipes
```

rake db:migrate

```
C:\Sites\receitasnet>rake db:migrate
== 20191009145025 AddNewFieldsToRecipes: migrating =====
-- add_column(:recipes, :prepare_mode, :text)
  -> 3.5726s
-- add_column(:recipes, :cost, :decimal)
  -> 1.6178s
== 20191009145025 AddNewFieldsToRecipes: migrated (5.1945s) =====
```

```
rake db:migrate:status
```

```
C:\Sites\receitasnet>rake db:migrate:status

database: receitasnet_development

  Status  Migration ID      Migration Name
  -----
    up      20191008220014  Create recipes
    up      20191009145025  Add new fields to recipes
```

Inserindo dados, pelo console, nos campos adicionados

```
rails c
```

```
reload!
```

```
r = Recipe.find_by(name: "Strogonoff de carne")
```

```
r.prepare_mode="Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne e deixe cozinhar por 20 minutos. Acrescente o ketchup e deixe cozinhar até obter um molho consistente e cremoso. Desligue o fogo e acrescente o creme de leite sem soro. Mexa até incorporar o molho ao creme. Coloque em uma forma refratária e decore com tempero e batata palha."
```

```
r.cost="45.00"
```

```
r.save
```

```
irb(main):006:0> r.save
  (0.6ms)  BEGIN
  Recipe Update (34.3ms)  UPDATE `recipes` SET `updated_at` = '2019-10-09 15:20:22', `prepare_mode` =
  'Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne e deixe cozinhar po
r 20 minutos. Acrescente o ketchup e deixe cozinhar até obter um molho consistente e cremoso. Desligu
e o fogo e acrescente o creme de leite sem soro. Mexa até incorporar o molho ao creme. Coloque em uma
 forma refratária e decore com tempero e batata palha.', `cost` = 45 WHERE `recipes`.`id` = 1
  (191.1ms)  COMMIT
=> true
```

```
r = Recipe.find_by(name: "Filé de frango à parmegiana")
```

```
r.prepare_mode="Tempere os filés de frango com o alho, suco de limão, ervas finas e sal à
gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo não muito quente. Retire
e deixe descansar. Coloque os filés em um refratário. Bubra com a mussarela em fatias.
Regue com o molho e leve ao forno bem quente, somente para derreter a mussarela. Retire
e sirva."
```

```
r.cost="50.00"
```

```
r.save
```

```
irb(main):010:0> r = Recipe.find_by(name: "Filé de frango à parmegiana")
  Recipe Load (0.9ms)  SELECT `recipes`.* FROM `recipes` WHERE `recipes`.`name` = 'Filé de frango à
parmegiana' LIMIT 1
=> #<Recipe id: 2, name: "Filé de frango à parmegiana", stuff: "1/2 kg de filé de frango, 2 ovos batidos com 1 pit...", calories: 160, created_at: "2019-10-09 01:23:38", updated_at: "2019-10-09 01:23:38", prepare_mode: nil, cost: nil>
< molho e leve ao forno bem quente, somente para derreter a mussarela. Retire e sirva."
=> "Tempere os filé de frango com o alho, suco de limão, ervas finas e sal à gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo não muito quente. Retire e deixe descansar. Coloque os filé s em um refratário. Bubra com a mussarela em fatias. Regue com o molho e leve ao forno bem quente, somente para derreter a mussarela. Retire e sirva."
irb(main):012:0> r.cost="50.00"
=> "50.00"
irb(main):013:0> r.save
  (0.6ms)  BEGIN
    Recipe Update (58.8ms)  UPDATE `recipes` SET `updated_at` = '2019-10-09 15:32:48', `prepare_mode` =
      'Tempere os filé de frango com o alho, suco de limão, ervas finas e sal à gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo não muito quente. Retire e deixe descansar. Coloque os filé s em um refratário. Bubra com a mussarela em fatias. Regue com o molho e leve ao forno bem quente, somente para derreter a mussarela. Retire e sirva.', `cost` = 50 WHERE `recipes`.`id` = 2
    (106.2ms)  COMMIT
=> true
```

```
r = Recipe.find_by(name: "Salmão ao forno")
```

```
r.update(prepare_mode:"Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180º a 200ºC, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga", cost: 60.00)
```

```
irb(main):014:0> r = Recipe.find_by(name: "Salmão ao forno")
  Recipe Load (2.9ms)  SELECT `recipes`.* FROM `recipes` WHERE `recipes`.`name` = 'Salmão ao forno'
LIMIT 1
=> #<Recipe id: 4, name: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite ", calories: 110, created_at: "2019-10-09 02:34:04", updated_at: "2019-10-09 02:34:04", prepare_mode: nil, cost: nil>
<Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga", cost: 60.00)
  (0.5ms)  BEGIN
    Recipe Update (60.0ms)  UPDATE `recipes` SET `updated_at` = '2019-10-09 15:42:17', `prepare_mode` =
      'Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180º a 200ºC, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga', `cost` = 60 WHERE `recipes`.`id` = 4
    (67.7ms)  COMMIT
=> true
```

receitasnet_development.recipes: 3 registros totais (aproximadamente)								Próximo	Mostrar todos	Ordem	Colunas (8/8)	Filtro
Id	Nome	Stuff	Calorias	Criado_em	Atualizado_em	Modo_de_preparo	Custo					
1	Stroganoff de carne	1 kg de carne, 3 cebolas médias, sal e pimenta-do...	100	2019-10-11 23:42:27	2019-10-12 00:08:08	Em uma panela, adicione o óleo, a carne, a cebol...	45					
2	Filé de frango à parmegiana	1/2 kg de filé de frango, 2 ovos batidos com 1 pit...	160	2019-10-11 23:43:00	2019-10-12 00:14:24	Tempere os filés de frango com o alho, suco de li...	50					
3	Salmão ao forno	1 salmão, sal grosso e azeite	110	2019-10-11 23:43:20	2019-10-12 00:15:26	Retire as espinhas do filé, passe o sal grosso e a...	60					

app/views/recipes/index.html.erb

```
<h1>Receitas</h1>

<ul>
  <% @recipes.each do |recipe| %>
    <li>Nome: <%= recipe.name %></li>
    <li>Ingredientes: <%= recipe.stuff %></li>
    <li>Calorias: <%= recipe.calories %></li>
    <li>Modo de preparo: <%= recipe.prepare_mode %></li>
    <li>Custo: <%= recipe.cost %></li>
    <hr>
  <% end %>
</ul>
```



- ## Receitas
- Nome: Strogonoff de carne
 - Ingredientes: 1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup
 - Calorias: 100
 - Modo de preparo: Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne e deixe cozinhar por 20 minutos. Acrescente o ketchup e deixe cozinhar até obter um molho consistente e cremoso. Desligue o fogo e acrescente o creme de leite sem soro. Mexa até incorporar o molho ao creme. Coloque em uma forma refratária e decore com tempero e batata palha.
 - Custo: 45

 - Nome: Filé de frango à parmegiana
 - Ingredientes: 1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 xícaras de chá de farinha de rosca, 1 caixinha de molho de tomate pronto, mussarela
 - Calorias: 160
 - Modo de preparo: Tempere os filés de frango com o alho, suco de limão, ervas finas e sal à gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo não muito quente. Retire e deixe descansar. Coloque os filés em um refratário. Bubra com a mussarela em fatias. Regue com o molho e leve ao forno bem quente, somente para derreter a mussarela. Retire e sirva.
 - Custo: 50

 - Nome: Salmão ao forno
 - Ingredientes: 1 salmão, sal grosso e azeite
 - Calorias: 110
 - Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga
 - Custo: 60

app/views/recipes/index.html.erb

```
<% @recipes.each do |recipe| %>
<article>
<header>
<h2><%= recipe.name %></h2>
</header>
<p><strong>Ingredientes: </strong><%= recipe.stuff %></p>
<p><strong>Calorias: </strong><%= recipe.calories %></p>
<p><strong>Modo de preparo: </strong><%= recipe.prepare_mode %></strong></p>
<p><strong>Custo: </strong><%= number_to_currency(recipe.cost) %></strong></p>
</article>
<% end %>
```

Strogonoff de carne

Ingredientes: 1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup

Calorias: 100

Modo de preparo: Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne e deixe cozinhar por 20 minutos. Acrescente o ketchup e deixe cozinhar até obter um molho consistente e cremoso. Desligue o fogo e acrescente o creme de leite sem soro. Mexa até incorporar o molho ao creme. Coloque em uma forma refratária e decore com tempero e batata palha.

Custo: \$45.00

Filé de frango á parmegiana

Ingredientes: 1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 xícaras de chá de farinha de rosca, 1 caixinha de molho de tomate pronto, mussarela

Calorias: 160

Modo de preparo: Tempere os filetes de frango com o alho, suco de limão, ervas finas e sal à gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo não muito quente. Retire e deixe descansar. Coloque os filetes em um refratário. Bubra com a mussarela em fatias. Regue com o molho e leve ao forno bem quente, somente para derreter a mussarela. Retire e sirva.

Custo: \$50.00

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 110

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo: \$60.00

ADICIONANDO AS ALTERAÇÕES NO GIT

git status

```
C:\Sites\receitasnet>git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   app/views/recipes/index.html.erb
    modified:   db/schema.rb

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  #
  db/migrate/20191009145025 add new fields to recipes.rb

no changes added to commit (use "git add" and/or "git commit -a")
```

git add .

```
C:\Sites\receitasnet>git add .
warning: LF will be replaced by CRLF in db/migrate/20191009145025 add new fields to recipes.rb.
The file will have its original line endings in your working directory
```

git status

```
C:\Sites\receitasnet>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   #
    modified:   app/views/recipes/index.html.erb
    new file:   db/migrate/20191009145025 add new fields to recipes.rb
    modified:   db/schema.rb
```

git commit -m "Inserindo novos campos na tabela recipes"

```
C:\Sites\receitasnet>git commit -m "Inserindo novos campos na tabela recipes"
[master 8e1ac38] Inserindo novos campos na tabela recipes
4 files changed, 20 insertions(+), 11 deletions(-)
create mode 100644 #
rewrite app/views/recipes/index.html.erb (100%)
create mode 100644 db/migrate/20191009145025 add new fields to recipes.rb
```

```
git push heroku master
```

Adicionando os campos na tabela do heroku

```
heroku run rake db:migrate
```

```
C:\Sites\receitasnet>heroku run rake db:migrate
Running `rake db:migrate` on ⬤ receitasnet-robertopinheiro... up, run.4319 (Free)
D, [2019-10-09T16:42:34.207414 #4] DEBUG -- :   (1.1ms)  SELECT pg_try_advisory_lock(5279828704805601030)
D, [2019-10-09T16:42:34.220983 #4] DEBUG -- :   (1.5ms)  SELECT "schema_migrations"."version" FROM "schema_migrations" ORDER BY "schema_migrations"."version" ASC
I, [2019-10-09T16:42:34.222095 #4]  INFO -- : Migrating to AddNewFieldsToRecipes (20191009145025)
D, [2019-10-09T16:42:34.224890 #4] DEBUG -- :   (1.0ms)  BEGIN
== 20191009145025 AddNewFieldsToRecipes: migrating =====
-- add_column(:recipes, :prepare_mode, :text)
D, [2019-10-09T16:42:34.227115 #4] DEBUG -- :   (1.8ms)  ALTER TABLE "recipes" ADD "prepare_mode" text
   -> 0.0021s
-- add_column(:recipes, :cost, :decimal)
D, [2019-10-09T16:42:34.228820 #4] DEBUG -- :   (1.4ms)  ALTER TABLE "recipes" ADD "cost" decimal
   -> 0.0017s
== 20191009145025 AddNewFieldsToRecipes: migrated (0.0040s) =====
D, [2019-10-09T16:42:34.234865 #4] DEBUG -- :   ActiveRecord::SchemaMigration Create (2.1ms)  INSERT INTO "schema_migrations" ("version") VALUES ($1) RETURNING "version"  [{"version": "20191009145025"}]
D, [2019-10-09T16:42:34.237511 #4] DEBUG -- :   (2.4ms)  COMMIT
D, [2019-10-09T16:42:34.245452 #4] DEBUG -- :   ActiveRecord::InternalMetadata Load (1.2ms)  SELECT "ar_internal_metadata".* FROM "ar_internal_metadata" WHERE "ar_internal_metadata"."key" = $1 LIMIT $2  [{"key": "environment"}, {"LIMIT": 1}]
D, [2019-10-09T16:42:34.255119 #4] DEBUG -- :   (1.0ms)  BEGIN
D, [2019-10-09T16:42:34.256852 #4] DEBUG -- :   (1.1ms)  COMMIT
D, [2019-10-09T16:42:34.258156 #4] DEBUG -- :   (1.1ms)  SELECT pg_advisory_unlock(5279828704805601030)
```

```
heroku run console
```

```
r = Recipe.find_by(name: "Salmão ao forno")
```

r.update(prepare_mode:"Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180º a 200º, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga", cost: 60.00)

```
C:\Sites\receitasnet>heroku run console
Running `console` on ⬤ receitasnet-robertopinheiro... up, run.2961 (Free)
Loading production environment (Rails 5.2.3)
irb(main):001:0> r = Recipe.find_by(name: "Salmão ao forno")
D, [2019-10-09T16:48:24.928804 #4] DEBUG -- :   Recipe Load (1.7ms)  SELECT "recipes".* FROM "recipes" WHERE "recipes"."name" = $1 LIMIT $2  [{"name": "Salmão ao forno"}, {"LIMIT": 1}]
=> #<Recipe id: 1, name: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite", calories: 110, created_at: "2019-10-09 02:54:53", updated_at: "2019-10-09 02:54:53", prepare_mode: nil, cost: nil>
<utros, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga", cost: 60.00)
D, [2019-10-09T16:48:44.024493 #4] DEBUG -- :   (1.1ms)  BEGIN
D, [2019-10-09T16:48:44.028200 #4] DEBUG -- :   Recipe Update (1.6ms)  UPDATE "recipes" SET "updated_at" = $1, "prepare_mode" = $2, "cost" = $3 WHERE "recipes"."id" = $4  [{"updated_at": "2019-10-09 16:48:44.025347"}, {"prepare_mode": "Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180º a 200º, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga"}, {"cost": "60.0"}, {"id": 1}]
D, [2019-10-09T16:48:44.030727 #4] DEBUG -- :   (2.2ms)  COMMIT
=> true
```

```
heroku open
```



AULA 07 – HELPERS

Helpers são métodos em Ruby que a view chama quando precisa gerar uma saída para o usuário.

O Rails já vem com vários métodos próprios. Um deles é o pluralize.

```
rails c
```

```
helper.pluralize(3, "Receita", "Receitas")
```

```
C:\Sites\receitasnet>rails c
Loading development environment (Rails 5.2.3)
irb(main):001:0> helper.pluralize(3, "Receita", "Receitas")
=> "3 Receitas"
irb(main):002:0> helper.pluralize(1, "Receita", "Receitas")
=> "1 Receita"
irb(main):003:0>
```

app/views/recipes/index.html.erb

```
<h1><%= pluralize(@recipes.size, "Receita", "Receitas") %></h1>
<% @recipes.each do |recipe| %>
<article>
<header>
<h2><%= recipe.name %></h2>
</header>
<p><strong>Ingredientes: </strong><%= truncate(recipe.stuff, length:40, separator:' ') %></p>
<p><strong>Calorias: </strong><%= formata_caloria(recipe) %></p>
<p><strong>Modo de preparo: </strong><%= recipe.prepare_mode %></strong></p>
<p><strong>Custo: </strong><%= number_to_currency(recipe.cost) %></strong></p>
</article>
<% end %>
```

app/helpers/recipes_helper.rb

```
module RecipesHelper
  def formata_caloria(recipe)
    if recipe.light?
      content_tag(:strong, "Light (menos de 100 calorias)")
    else
      recipe.calories
    end
  end
end
```

app/models/recipe.rb

```
class Recipe < ApplicationRecord
  def light?
    calories < 100
  end
end
```

Criando uma nova receita (light)

rails c

Recipe.create(name:"Salada de peito de peru", stuff:"1 maço de alface(ou rúcula), suco de limão, 200g de peito de peru defumado, 1 pote (120g) de cream cheese light, 2 colheres (chá) de azeite de oliva extra virgem, 1 colher (chá) de aceto bolsômico (ou vinagre), sal a gosto", calories:95, prepare_mode:"Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e temprere a salada", cost:20.00)

```
Recipe Create (26.1ms)  INSERT INTO `recipes` (`name`, `stuff`, `calories`, `created_at`, `updated_at`, `prepare_mode`, `cost`) VALUES ('Salada de peito de peru', '1 maço de alface(ou r\xFcula), suco de lim\xF3o, 200g de peito de peru defumado, 1 pote (120g) de cream cheese light, 2 colheres (ch\xE1) de azeite de oliva extra virgem, 1 colher (ch\xE1) de aceto bols\xF3mico (ou vinagre), sal a gosto', 95, '2019-10-09 20:38:50', '2019-10-09 20:38:50', 'Lave a alface (ou r\xFcula) e arrume em uma travessa. Regue com o suco de lim\xF3o. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e temprere a salada', 20)
(125.9ms)  COMMIT
=> #<Recipe id: 5, name: "Salada de peito de peru", stuff: "1 ma\x87o de alface(ou r\xA3cula), suco d e lim\xC6o, 200g d...", calories: 95, created_at: "2019-10-09 20:38:50", updated_at: "2019-10-09 20:38:50", prepare_mode: "Lave a alface (ou r\xA3cula) e arrume em uma travessa...", cost: 20>
irb(main):004:0>
```

receitasnet_development.recipes: 4 registros totais (aproximadamente)						
	id	name	stuff	calories	created_at	updated_at
1	1	Strogonoff de carne	1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto...	100	2019-10-09 00:45:45	2019-10-09 15:20:22
2	2	Filé de frango à parmegiana	1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 ...	160	2019-10-09 01:23:38	2019-10-09 15:32:48
4	4	Salmão ao forno	1 salmão, sal grosso e azeite	110	2019-10-09 02:34:04	2019-10-09 15:42:17
5	5	Salada de peito de peru	1 maço de alface (ou rúcula), suco de limão, 200g de peito de ...	95	2019-10-09 20:38:50	2019-10-09 20:38:50

4 Receitas

Strogonoff de carne

Ingredientes: 1 kg de carne, 3 cebolas médias, sal...

Calorias: 100

Modo de preparo: Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne e deixe cozinhar por 20 minutos. Acrescente o ketchup e deixe cozinhar até obter um molho consistente e cremoso. Desligue o fogo e acrescente o creme de leite sem soro. Mexa até incorporar o molho ao creme. Coloque em uma forma refratária e decore com tempero e batata palha.

Custo: \$45.00

Filé de frango à parmegiana

Ingredientes: 1/2 kg de filé de frango, 2 ovos...

Calorias: 160

Modo de preparo: Tempere os filetes de frango com o alho, suco de limão, ervas finas e sal à gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo não muito quente. Retire e deixe descansar. Coloque os filetes em um refratário. Bubra com a mussarela em fatias. Regue com o molho e leve ao forno bem quente, sorridente para derreter a mussarela. Retire e sirva.

Custo: \$50.00

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 110

Modo de preparo: Retire as espinhas do filete, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo: \$60.00

Salada de peito de peru

Ingredientes: 1 maço de alface (ou rúcula), suco de...

Calorias: Light (menos de 100 calorias)

Modo de preparo: Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada

Custo: \$20.00

ADICIONANDO AS ALTERAÇÕES NO GIT

git add .

```
C:\Sites\receitasnet>git add .
warning: LF will be replaced by CRLF in app/helpers/recipes_helper.rb.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in app/models/recipe.rb.
The file will have its original line endings in your working directory
```

git status

```
C:\Sites\receitasnet>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   app/helpers/recipes_helper.rb
    modified:   app/models/recipe.rb
    modified:   app/views/recipes/index.html.erb
```

```
git commit -m "Criando helpers"
```

```
C:\Sites\receitasnet>git commit -m "Criando helpers"
[master 530e6e4] Criando helpers
 3 files changed, 13 insertions(+), 2 deletions(-)
```

```
git log
```

```
C:\Sites\receitasnet>git log
commit 7a692ee1bc29370e28281d8eeccdc94791d3d813 (HEAD -> master, heroku/master)
Author: Roberto Pinheiro <betopinheiro1005@yahoo.com.br>
Date:   Fri Oct 11 21:57:25 2019 -0300

    Criando helpers

commit 1119b8f554d5593610fca062d44bc644fca4748c
Author: Roberto Pinheiro <betopinheiro1005@yahoo.com.br>
Date:   Fri Oct 11 21:19:59 2019 -0300

    Inserindo novos campos na tabela recipes

commit 388ff89c6b1326946a1f594856507a574e006aff
Author: Roberto Pinheiro <betopinheiro1005@yahoo.com.br>
Date:   Fri Oct 11 20:52:33 2019 -0300

    Retira hardcoded do controller e ajusta a view

commit 41ed22ee0a60d8ba1bebbacad2c5363e717c7215
Author: Roberto Pinheiro <betopinheiro1005@yahoo.com.br>
Date:   Fri Oct 11 20:45:51 2019 -0300

    Criando model Recipe

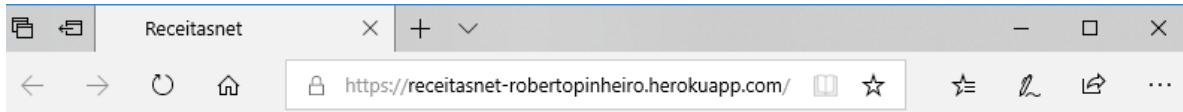
commit 354919f6cf0a1f071b95e6add80da654e6380dc0
Author: Roberto Pinheiro <betopinheiro1005@yahoo.com.br>
Date:   Fri Oct 11 20:34:16 2019 -0300

    Inserindo a rota raiz

commit a11b48f30ca770cbc3b409ab6268102aa3c19c08
Author: Roberto Pinheiro <betopinheiro1005@yahoo.com.br>
Date:   Fri Oct 11 20:17:44 2019 -0300

    Commit inicial
```

```
git push heroku master
```



Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 110

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

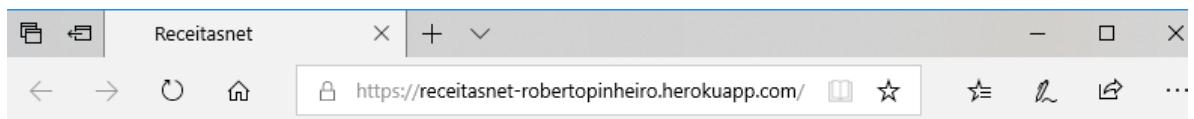
Custo: \$60.00

ADICIONANDO RECEITA LIGHT NO HEROKU

```
heroku run console
```

Recipe.create(name:"Salada de peito de peru", stuff:"1 maço de alface(ou rúcula), suco de limão, 200g de peito de peru defumado, 1 pote (120g) de cream cheese light, 2 colheres (chá) de azeite de oliva extra virgem, 1 colher (chá) de aceto bolsômico (ou vinagre), sal a gosto", calories:95, prepare_mode:"Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada", cost:20.00)

```
C:\Sites\receitasnet>heroku run console
Running console on • receitasnet-robertopinheiro... up, run.4574 (Free)
Loading production environment (Rails 5.2.3)
< o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada", cost:20.00)
D, [2019-10-09T21:02:12.549001 #4] DEBUG -- :   (1.4ms)  BEGIN
D, [2019-10-09T21:02:12.553078 #4] DEBUG -- :   Recipe Create (2.2ms)  INSERT INTO "recipes" ("name", "stuff", "calories", "created_at", "updated_at", "prepare_mode", "cost") VALUES ($1, $2, $3, $4, $5, $6, $7) RETURNING "id"
[[{"name": "Salada de peito de peru"}, {"stuff": "1 ma\u00e7o de alface(ou r\u00fccula), suco de lim\u00e3o, 200g de peito de peru defumado, 1 pote (120g) de cream cheese light, 2 colheres (ch\u00e1) de azeite de oliva extra virgem, 1 colher (ch\u00e1) de aceto bols\u00f3mico (ou vinagre), sal a gosto"}, {"calories": 95}, {"created_at": "2019-10-09 21:02:12.549001"}, {"updated_at": "2019-10-09 21:02:12.549350"}, {""prepare_mode": "Lave a alface (ou r\u00fccula) e arrume em uma travessa. Regue com o suco de lim\u00e3o. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada"}, {"cost": "20.0"}]
D, [2019-10-09T21:02:12.555587 #4] DEBUG -- :   (2.0ms)  COMMIT
=> #<Recipe id: 2, name: "Salada de peito de peru", stuff: "1 ma\u00e7o de alface(ou r\u00fccula), suco de lim\u00e3o, 200g d...", calories: 95, created_at: "2019-10-09 21:02:12", updated_at: "2019-10-09 21:02:12", prepare_mode: "Lave a alface (ou r\u00fccula) e arrume em uma travessa...", cost: 0.2e2>
```



2 Receitas

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 110

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo: \$60.00

Salada de peito de peru

Ingredientes: 1 maço de alface (ou rúcula), suco de...

Calorias: Light (menos de 100 calorias)

Modo de preparo: Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada

Custo: \$20.00

AULA 08 – CRIANDO O SHOW

`app/controllers/recipes_controller.rb`

```
class RecipesController < ApplicationController
  def index
    @recipes = Recipe.all
  end

  def show
    @recipe = Recipe.find(params[:id])
  end
end
```

`app/views/recipes/show.html.erb`

```
<article>
  <head>
    <h1><%= @recipe.name %></h1>
  </head>
  <h3>Ingredientes</h3>
  <p><%= @recipe.stuff %></p>
  <h3>Calorias</h3>
  <p><%= @recipe.calories %></p>
  <h3>Modo de preparo</h3>
  <p><%= @recipe.prepare_mode %></p>
  <h3>Custo</h3>
  <p><%= number_to_currency(@recipe.cost) %></p>
</article>
```



ADICIONANDO AS ALTERAÇÕES NO GIT

git add .

```
C:\Sites\receitasnet>git add .
warning: LF will be replaced by CRLF in app/controllers/recipes_controller.rb.
The file will have its original line endings in your working directory
```

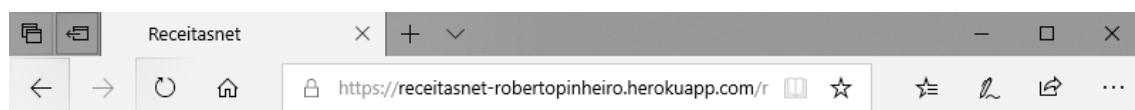
git status

```
C:\Sites\receitasnet>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   app/controllers/recipes_controller.rb
    new file:   app/views/recipes/show.html.erb
```

git commit -m "Criando o show"

```
C:\Sites\receitasnet>git commit -m "Criando o show"
[master 900cc51] Criando o show
 2 files changed, 17 insertions(+)
 create mode 100644 app/views/recipes/show.html.erb
```

git push heroku master
heroku open



Salmão ao forno

Ingredientes

1 salmão, sal grosso e azeite

Calorias

110

Modo de preparo

Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo

\$60.00

AULA 09 – CRIANDO OS LINKS

Para links usar as rotas nomeadas.

app/views/recipes/index.html.erb

```
<h1><%= pluralize(@recipes.size, "Receita", "Receitas") %></h1>
<% @recipes.each do |recipe| %>
<article>
<header>
<h2><%= link_to recipe.name, recipe %></h2>
</header>
<p><strong>Ingredientes: </strong><%= truncate(recipe.stuff, length:40, separator:' ') %></p>
<p><strong>Calorias: </strong><%= formata_caloria(recipe) %></p>
<p><strong>Modo de preparo: </strong><%= recipe.prepare_mode %></strong></p>
<p><strong>Custo: </strong><%= number_to_currency(recipe.cost) %></strong></p>
</article>
<% end %>
```

rake routes

```
receitasnet (master) $ rake routes
Prefix Verb URI Pattern Controller#Action
  root GET / recipes#index
  recipes GET /recipes(.:format) recipes#index
           POST /recipes(.:format) recipes#create
new_recipe GET /recipes/new(.:format) recipes#new
edit_recipe GET /recipes/:id/edit(.:format) recipes#edit
  recipe GET /recipes/:id(.:format) recipes#show
         PATCH /recipes/:id(.:format) recipes#update
         PUT /recipes/:id(.:format) recipes#update
    DELETE /recipes/:id(.:format) recipes#destroy
```

```
receitasnet (master) $ rails c
Running via Spring preloader in process 2351
Loading development environment (Rails 4.2.5.2)
2.2.1 :001 > app.recipes_path
=> "/recipes"
2.2.1 :002 > app.recipes_url
=> "http://www.example.com/recipes"
2.2.1 :003 > app.recipe_path(1)
=> "/recipes/1"
```

app/views/recipes/show.html.erb

```
<article>
  <head>
    <h1><%= @recipe.name %></h1>
  </head>
  <h3>Ingredientes</h3>
  <p><%= @recipe.stuff %></p>
  <h3>Calorias</h3>
  <p><%= @recipe.calories %></p>
  <h3>Modo de preparo</h3>
  <p><%= @recipe.prepare_mode %></p>
  <h3>Custo</h3>
  <p><%= number_to_currency(@recipe.cost) %></p>
</article>

<%= link_to "Receitas", recipes_path %>
```



4 Receitas

Strogonoff de carne

Ingredientes: 1 kg de carne, 3 cebolas médias, sal...

Calorias: 100

Modo de preparo: Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne e deixe cozinhar por 20 minutos. Acrescente o ketchup e deixe cozinhar até obter um molho consistente e cremoso. Desligue o fogo e acrescente o creme de leite sem soro. Mexa até incorporar o molho ao creme. Coloque em uma forma refratária e decore com tempero e batata palha.

Custo: \$45.00

Filé de frango á parmegiana

Ingredientes: 1/2 kg de filé de frango, 2 ovos...

Calorias: 160

Modo de preparo: Tempere os filetes de frango com o alho, suco de limão, ervas finas e sal à gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo não muito quente. Retire e deixe descansar. Coloque os filetes em um refratário. Bubra com a mussarela em fatias. Regue com o molho e leve ao forno bem quente, somente para derreter a mussarela. Retire e sirva.

Custo: \$50.00

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 110

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga.

Custo: \$60.00

Salada de peito de peru

Ingredientes: 1 maço de alface(ou rúcula), suco de...

The screenshot shows a web browser window with the title 'Receitasnet'. The address bar displays 'localhost:3000/recipes/1'. The main content area shows a recipe for 'Strogonoff de carne'. It includes sections for 'Ingredientes' (Ingredients), 'Calorias' (Calories), 'Modo de preparo' (Preparation Method), 'Custo' (Cost), and a link to 'Receitas' (Recipes). The preparation method describes cooking the meat, onions, tomatoes, and broth for 20 minutes, then adding ketchup and cream until a creamy consistency is achieved.

Strogonoff de carne

Ingredientes
1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup

Calorias
100

Modo de preparo
Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne e deixe cozinhar por 20 minutos. Acrescente o ketchup e deixe cozinhar até obter um molho consistente e cremoso. Desligue o fogo e acrescente o creme de leite sem soro. Mexa até incorporar o molho ao creme. Coloque em uma forma refratária e decore com tempero e batata palha.

Custo
\$45.00

[Receitas](#)

ADICIONANDO AS ALTERAÇÕES NO GIT

```
git add .
```

```
git status
```

```
git commit -m "Criando links no index e no show"
```

```
git push heroku master
```

```
heroku open
```



2 Receitas

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 110

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo: \$60.00

Salada de peito de peru

Ingredientes: 1 maço de alface(ou rúcula), suco de...

Calorias: Light (menos de 100 calorias)

Modo de preparo: Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada

Custo: \$20.00

The screenshot shows a Microsoft Edge browser window with the title bar "Receitasnet". The address bar displays the URL "https://receitasnet-robertopinheiro.herokuapp.com/recipes/1". The main content area of the browser shows a recipe card for "Salmão ao forno".

Salmão ao forno

Ingredientes

1 salmão, sal grosso e azeite

Calorias

110

Modo de preparo

Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo

\$60.00

[Receitas](#)

AULA 10 – UM POUCO DE RUBY

```
receitasnet (master) $ rails c
bash: raisl: command not found
receitasnet (master) $ rails c
Running via Spring preloader in process 634
Loading development environment (Rails 4.2.5.2)
2.2.1 :001 > 2+2
=> 4
2.2.1 :002 > "Olá Mundo"
=> "Olá Mundo"
2.2.1 :003 > "Olá Mundo".class
=> String
2.2.1 :004 > a= "Olá Mundo"
=> "Olá Mundo"
2.2.1 :005 > a
=> "Olá Mundo"
2.2.1 :006 > a= 'Olá Mundo'
=> "Olá Mundo"
2.2.1 :007 > 'I\'m a ruby developer'
=> "I'm a ruby developer"
2.2.1 :008 > "I'm a ruby developer"
=> "I'm a ruby developer"
2.2.1 :009 > "Olá" + " Mundo"
=> "Olá Mundo"
```

```
2.2.1 :010 > 2-1
=> 1
2.2.1 :011 > n=1
=> 1
2.2.1 :012 > n=2
=> 2
2.2.1 :013 > n.class
=> Fixnum
2.2.1 :014 > n2=3
=> 3
2.2.1 :015 > n + n2
=> 5
```

```
2.2.1 :001 > nome= "Fabricio"
=> "Fabricio"
2.2.1 :002 > curso="Ruby on rails"
=> "Ruby on rails"
2.2.1 :003 > "Eu #{nome.upcase} estou fazendo o curso de #{curso.capitalize} na codifike"
=> "Eu FABRICIO estou fazendo o curso de Ruby on rails na codifike"
2.2.1 :004 > curso="ruby on rails"
=> "ruby on rails"
2.2.1 :005 > "Eu #{nome.upcase} estou fazendo o curso de #{curso.capitalize} na codifike"
=> "Eu FABRICIO estou fazendo o curso de Ruby on rails na codifike"
```

```
2.2.1 :006 > a= 123.to_s
=> "123"
2.2.1 :007 > a.to_i
=> 123

2.2.1 :008 > "fabricio".object_id
=> 70282938597660
2.2.1 :009 > "fabricio".object_id
=> 70282920083980
2.2.1 :010 > :fabricio.object_id
=> 6894108
2.2.1 :011 > :fabricio.object_id
=> 6894108

2.2.1 :012 > "string".to_sym
=> :string
2.2.1 :013 > :symbol.to_s
=> "symbol"

2.2.1 :014 > frutas= [ 'maças', 'bananas', 'laranjas']
=> ["maças", "bananas", "laranjas"]
2.2.1 :015 > frutas= [ 'maças', 'bananas', 'laranjas', 1]
=> ["maças", "bananas", "laranjas", 1]
2.2.1 :016 > frutas= [ 'maças', 'bananas', 'laranjas', 1, ['morango']]
=> ["maças", "bananas", "laranjas", 1, ["morango"]]
2.2.1 :017 > frutas= [ 'maças', 'bananas', 'laranjas']
=> ["maças", "bananas", "laranjas"]
2.2.1 :018 > frutas[0]
=> "maças"
2.2.1 :019 > frutas[1]
=> "bananas"
2.2.1 :020 > frutas[2]
=> "laranjas"
2.2.1 :021 > frutas[4]
=> nil
2.2.1 :022 > frutas.first
=> "maças"
2.2.1 :023 > frutas.last
=> "laranjas"
2.2.1 :024 > frutas.length
=> 3
2.2.1 :025 > frutas.size
=> 3
2.2.1 :026 > frutas.empty?
=> false
2.2.1 :027 > frutas.include?('morango')
=> false
```

```
2.2.1 :028 > frutas.push('morango')
=> ["maças", "bananas", "laranjas", "morango"]
2.2.1 :029 > frutas << 2
=> ["maças", "bananas", "laranjas", "morango", 2]
2.2.1 :030 > frutas.insert(2,'kiwi')
=> ["maças", "bananas", "kiwi", "laranjas", "morango", 2]
2.2.1 :031 > frutas.pop
=> 2
2.2.1 :032 > frutas
=> ["maças", "bananas", "kiwi", "laranjas", "morango"]
2.2.1 :033 > frutas.delete('kiwi')
=> "kiwi"
2.2.1 :034 > frutas
=> ["maças", "bananas", "laranjas", "morango"]
2.2.1 :035 > num= [1,2,2,3,4,5,5]
=> [1, 2, 2, 3, 4, 5, 5]
2.2.1 :036 > num.uniq
=> [1, 2, 3, 4, 5]

2.2.1 :038 > frutas.each { |f| puts f, ' '}
maças
bananas
laranjas
morango

=> ["maças", "bananas", "laranjas", "morango"]
2.2.1 :039 > frutas.each { |f| puts f }
maças
bananas
laranjas
morango
=> ["maças", "bananas", "laranjas", "morango"]
2.2.1 :040 > frutas.each do |f|
2.2.1 :041 >     print f, ' '
2.2.1 :042?>   end
maças bananas laranjas morango => ["maças", "bananas", "laranjas", "morango"]
2.2.1 :043 > frutas.join
=> "maçasbananaslaranjasmorango"
```

```
2.2.1 :044 > frutas.join('-')
=> "maças-bananas-laranjas-morango"
2.2.1 :045 >
2.2.1 :046 >
2.2.1 :047 >
2.2.1 :048 >
2.2.1 :049 >   receitas = {nome:'Salmão', calorias: 110}
=> {:nome=>"Salmão", :calorias=>110}
2.2.1 :050 > print receitas[:nome]
Salmão => nil
2.2.1 :051 > receitas[:nome]='sopa'
=> "sopa"
2.2.1 :052 > receitas
=> {:nome=>"sopa", :calorias=>110}
2.2.1 :053 > receitas.each { |k, v| puts "Chave #{k} pertence ao valor #{v}" }
Chave nome pertence ao valor sopa
Chave calorias pertence ao valor 110
=> {:nome=>"sopa", :calorias=>110}
2.2.1 :054 > receitas.each do | k,v|
2.2.1 :055 >   puts "Chave #{k} pertence ao valor #{v}"
2.2.1 :056?> end
Chave nome pertence ao valor sopa
Chave calorias pertence ao valor 110
=> {:nome=>"sopa", :calorias=>110}
```

```
2.2.1 :057 > receitas.empty?
=> false
2.2.1 :058 > receitas.has_key?(:nome)
=> true
2.2.1 :059 > receitas.has_value?('sopa')
=> true
2.2.1 :060 > receitas.has_value?('churrasco')
=> false
2.2.1 :061 > receitas.length
=> 2
2.2.1 :062 > receitas.delete(:nome)
=> "sopa"
2.2.1 :063 > receitas
=> {:calorias=>110}
2.2.1 :064 > (0..5).to_a
=> [0, 1, 2, 3, 4, 5]
```

```
2.2.1 :065 >
2.2.1 :066 >
2.2.1 :067 >
2.2.1 :068 >     def maior(num)
2.2.1 :069?>     if num > 10
2.2.1 :070?>         puts "maior que 10"
2.2.1 :071?>     else
2.2.1 :072 >         puts "menor que 10"
2.2.1 :073?>     end
2.2.1 :074?>     end
=> :maior
2.2.1 :075 > maior(5)
menor que 10
=> nil
2.2.1 :076 > maior(20)
maior que 10
=> nil
```

AULA 11 – FORMULÁRIOS DE EDIÇÃO

`app/controllers/recipes_controller.rb`

```
class RecipesController < ApplicationController
  def index
    @recipes = Recipe.all
  end

  def show
    @recipe = Recipe.find(params[:id])
  end

  def edit
    @recipe = Recipe.find(params[:id])
  end

  def update
    @recipe = Recipe.find(params[:id])
    recipe_params = params.require(:recipe).permit(:name, :stuff, :calories, :prepare_mode, :cost)
    @recipe.update(recipe_params)
    redirect_to @recipe
  end

end
```

app/views/recipes/edit.html.erb

```
<%= form_for(@recipe) do |f| %>
<p>
  <%= f.label :Nome %><br>
  <%= f.text_field :name %>
</p>
<p>
  <%= f.label :Ingredientes %><br>
  <%= f.text_area :stuff, cols: 40, rows: 5 %>
</p>
<p>
  <%= f.label :Calorias %><br>
  <%= f.number_field :calories %>
</p>
<p>
  <%= f.label :Modo_de_Prepargo %><br>
  <%= f.text_area :prepare_mode, cols: 40, rows: 5 %>
</p>
<p>
  <%= f.label :Custo %><br>
  <%= f.number_field :cost %>
</p>
<p>
  <%= f.submit "Salvar receita" %><br><br>
  <%= link_to "Cancelar alterações", root_path %>
</p>
<% end %>
```

localhost:3000/recipes/1/edit

Apps Gerador de Formul... Formulário com vali...

Nome
Strogonoff de carne

Ingredientes
1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup

Calorias
100

Modo de preparo
Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne e deixe cozinhar por 20 minutos. Acrescente o ketchup e deixe cozinhar até obter um molho

Custo
45

[Salvar receita](#)

[Cancelar alterações](#)



Strogonoff de carne

Ingredientes
1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino à gosto, 2 latas de creme de leite, 2 colheres (de sopa) de ketchup

Calorias
150

Modo de preparo
Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne e deixe cozinhar por 20 minutos. Acrescente o ketchup e deixe cozinhar até obter um molho consistente e cremoso. Desligue o fogo e acrescente o creme de leite sem soro. Mexa até incorporar o molho ao creme. Coloque em uma forma refratária e decore com tempero e batata palha.

Custo
\$45.00

[Receitas](#) [Editar Receita](#)

ADICIONANDO AS ALTERAÇÕES NO GIT

```
git add .
git status
git commit -m "Criando formulario de edicao"
```

```
C:\Sites\receitasnet>git add .
warning: LF will be replaced by CRLF in app/controllers/recipes_controller.rb.
The file will have its original line endings in your working directory

C:\Sites\receitasnet>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   app/controllers/recipes_controller.rb
    new file:   app/views/recipes/edit.html.erb
    modified:   app/views/recipes/show.html.erb

C:\Sites\receitasnet>
C:\Sites\receitasnet>git commit -m "Criando formulário de edição"
[master 73009c0] Criando formulário de edição
 3 files changed, 40 insertions(+), 1 deletion(-)
 create mode 100644 app/views/recipes/edit.html.erb
```

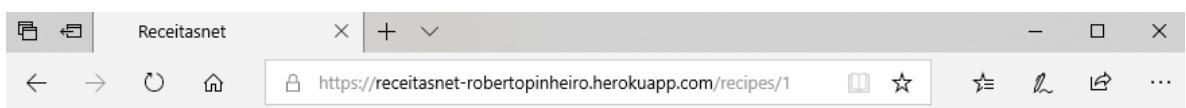
```
git push heroku master
```

```
heroku open
```

The screenshot shows a web browser window with the title 'Receitasnet'. The URL in the address bar is <https://receitasnet-robertopinheiro.herokuapp.com/recipes/1/edit>. The page displays a form for editing a recipe:

- Nome:** Salmão ao forno
- Ingredientes:** 1 salmão, sal grosso e azeite
- Calorias:** 115
- Modo de preparo:** Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar.
- Custo:** 60.0

At the bottom of the form are two buttons: 'Salvar receita' (Save recipe) and 'Cancelar alterações' (Cancel changes).



Salmão ao forno

Ingredientes

1 salmão, sal grosso e azeite

Calorias

115

Modo de preparo

Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo

\$60.00

[Receitas](#)

AULA 12 – FORMULÁRIOS DE CRIAÇÃO

Criando o link para gerar uma nova receita

app/views/recipes/index.html.erb

```
<h1><%= pluralize(@recipes.size, "Receita", "Receitas") %></h1>
<% @recipes.each do |recipe| %>
  <article>
    <header>
      <h2><%= link_to recipe.name, recipe %></h2>
    </header>
    <p><strong>Ingredientes: </strong><%= truncate(recipe.stuff, length:40, separator:' ') %></p>
    <p><strong>Calorias: </strong><%= formata_caloria(recipe) %></p>
    <p><strong>Modo de preparo: </strong><%= recipe.prepare_mode %></strong></p>
    <p><strong>Custo: </strong><%= number_to_currency(recipe.cost) %></strong></p>
  </article>
<% end %>
<br>
<%= link_to 'Nova receita', new_recipe_path %>
```

4 Receitas

Strogonoff de carne

Ingredientes: 1 kg de carne, 3 cebolas médias, sal...

Calorias: 100

Modo de preparo: Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne e deixe cozinhar por 20 minutos. Acrescente o ketchup e deixe cozinhar até obter um molho consistente e cremoso. Desligue o fogo e acrescente o creme de leite sem soro. Mexa até incorporar o molho ao creme. Coloque em uma forma refratária e decore com tempero e batata palha.

Custo: \$45.00

Filé de frango à parmegiana

Ingredientes: 1/2 kg de filé de frango, 2 ovos...

Calorias: 160

Modo de preparo: Tempere os filés de frango com o alho, suco de limão, ervas finas e sal à gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo não muito quente. Retire e deixe descansar. Coloque os filés em um refratário. Bubra com a mussarela em fatias. Regue com o molho e leve ao forno bem quente, somente para derreter a mussarela. Retire e sirva.

Custo: \$50.00

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 110

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo: \$60.00

Salada de peito de peru

Ingredientes: 1 maço de alface(ou rúcula), suco de...

Calorias: Light (menos de 100 calorias)

Modo de preparo: Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada

Custo: \$20.00

[Nova receita](#)

app/controllers/recipes_controller.rb

```
class RecipesController < ApplicationController
  def index
    @recipes = Recipe.all
  end

  def show
    @recipe = Recipe.find(params[:id])
  end

  def new
    @recipe = Recipe.new
  end

  def create
    recipe_params = params.require(:recipe).permit(:name, :stuff, :calories, :prepare_mode, :cost)
    @recipe = Recipe.new(recipe_params)
    if @recipe.save
      redirect_to @recipe
    else
      render :new
    end
  end

  def edit
    @recipe = Recipe.find(params[:id])
  end

  def update
    @recipe = Recipe.find(params[:id])
    recipe_params = params.require(:recipe).permit(:name, :stuff, :calories, :prepare_mode, :cost)
    @recipe.update(recipe_params)
    redirect_to @recipe
  end
end
```

← → ⌛ ⓘ localhost:3000/recipes/new

_apps Gerador de Formul... Formulário com vali...

Nome

Ingredientes

Calorias

Modo de preparo

Custo

[Cancelar alterações](#)

Nome: Fettuccine

Ingredientes:

250g de Fettuccine, 70g bacon, 2 caixas de creme de leite, 3 colheres (sopa) de queijo ralado de saquinho (parmesão), 2 dentes de alho, 1 colher (de sopa) de manteiga/margarina, sal a gosto.

Calorias: 280

Modo de preparo:

Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.

Custo: 50.00

localhost:3000/recipes/new

Apps Gerador de Formul... Formulário com vali...

Nome
Fettuccine

Ingredientes

```
250g de Fettuccine, 70g bacon, 2 caixas de creme de leite, 3 colheres (sopa) de queijo ralado de saquinho (parmesão), 2 dentes de alho, 1 colher (de sopa) de manteiga/margarina, sal a gosto.
```

Calorias
280

Modo de preparo

```
até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.
```

Custo
50.00

Salvar receita

localhost:3000/recipes/5

Fettuccine

Ingredientes

250g de Fettuccine, 70g bacon, 2 caixas de creme de leite, 3 colheres (sopa) de queijo ralado de saquinho (parmesão), 2 dentes de alho, 1 colher (de sopa) de manteiga/margarina, sal a gosto.

Calorias

280

Modo de preparo

Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.

Custo

\$50.00

[Receitas](#)

ADICIONANDO AS ALTERAÇÕES NO GIT

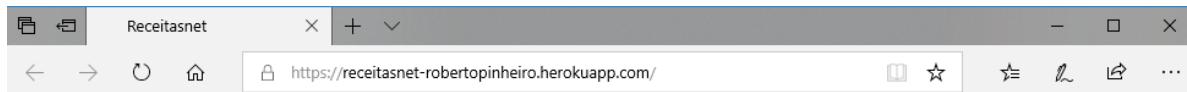
```
git add .
git status
git commit -m "Criando formulario de criacao"
```

```
C:\Sites\receitasnet>git add .
warning: LF will be replaced by CRLF in app/controllers/recipes_controller.rb.
The file will have its original line endings in your working directory

C:\Sites\receitasnet>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   app/controllers/recipes_controller.rb
    modified:   app/views/recipes/index.html.erb
    new file:   app/views/recipes/new.html.erb

C:\Sites\receitasnet>git commit -m "Criando formulário de criação"
[master b286078] Criando formulário de criação
 3 files changed, 42 insertions(+), 1 deletion(-)
 create mode 100644 app/views/recipes/new.html.erb
```

```
git push heroku master
heroku open
```



2 Receitas

Salada de peito de peru

Ingredientes: 1 maço de alface(ou rúcula), suco de...

Calorias: Light (menos de 100 calorias)

Modo de preparo: Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada

Custo: \$20.00

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 115

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo: \$60.00

[Nova receita](#)

Nome: Fettuccine

Ingredientes:

250g de Fettuccine, 70g bacon, 2 caixas de creme de leite, 3 colheres (sopa) de queijo ralado de saquinho (parmesão), 2 dentes de alho, 1 colher (de sopa) de manteiga/margarina, sal a gosto.

Calorias: 280

Modo de preparo:

Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.

Custo: 50.00

← → C ① localhost:3000/recipes/new

Criando receitas

Nome

Fettuccine

Ingredientes

250g de Fettuccine, 70g bacon, 2 caixas de creme de leite, 3 colheres (sopa) de queijo ralado de saquinho (parmesão), 2 dentes de alho, 1 colher (de sopa) de manteiga/margarina, sal a gosto.

Calorias

280

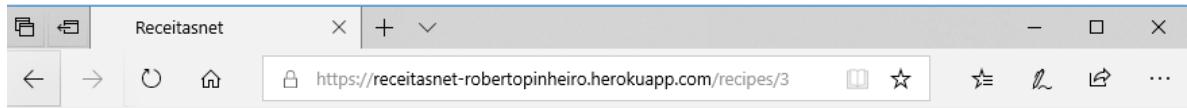
Modo de preparo

até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.

Custo

50.00

[Cancelar alterações](#)



Fettuccine

Ingredientes

250g de Fettuccine, 70g bacon, 2 caixas de creme de leite, 3 colheres (sopa) de queijo ralado de saquinho (parmesão), 2 dentes de alho, 1 colher (de sopa) de manteiga/margarina, sal a gosto.

Calorias

280

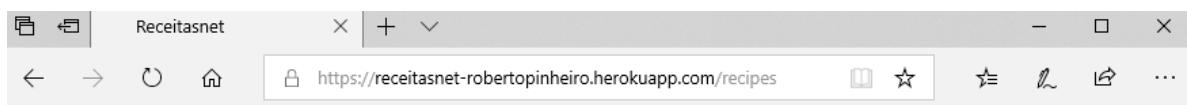
Modo de preparo

Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.

Custo

\$50.00

[Receitas](#)



3 Receitas

Salada de peito de peru

Ingredientes: 1 maço de alface(ou rúcula), suco de...

Calorias: Light (menos de 100 calorias)

Modo de preparo: Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada

Custo: \$20.00

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 115

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo: \$60.00

Fettuccine

Ingredientes: 250g de Fettuccine, 70g bacon, 2...

Calorias: 280

Modo de preparo: Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.

Custo: \$50.00

[Nova receita](#)

AULA 13 – DELETANDO RECEITAS

app/controllers/recipes_controller.rb

```
class RecipesController < ApplicationController
  def index
    @recipes = Recipe.all
  end

  def show
    @recipe = Recipe.find(params[:id])
  end

  def new
    @recipe = Recipe.new
  end

  def create
    recipe_params = params.require(:recipe).permit(:name, :stuff, :calories, :prepare_mode, :cost)
    @recipe = Recipe.new(recipe_params)
    @recipe.save
    redirect_to @recipe
  end

  def edit
    @recipe = Recipe.find(params[:id])
  end

  def update
    @recipe = Recipe.find(params[:id])
    recipe_params = params.require(:recipe).permit(:name, :stuff, :calories, :prepare_mode, :cost)
    @recipe.update(recipe_params)
    redirect_to @recipe
  end

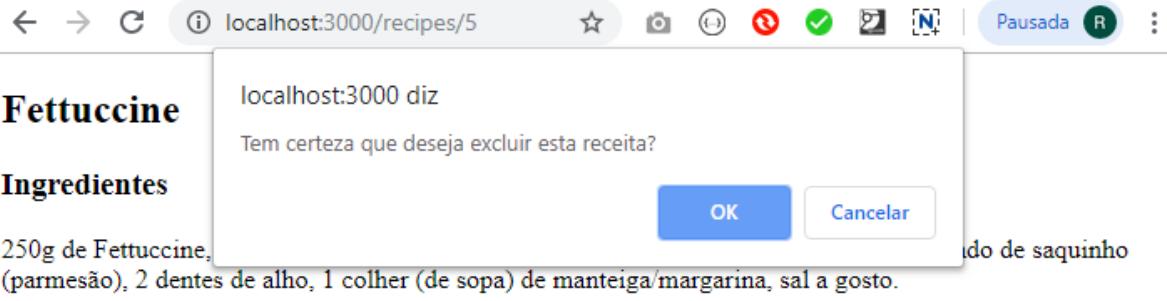
  def destroy
    @recipe = Recipe.find(params[:id])
    @recipe.destroy
    redirect_to recipes_url
  end
end
```

app/views/recipes/show.html.erb

```
<article>
  <head>
    <h1><%= @recipe.name %></h1>
  </head>
  <h3>Ingredientes</h3>
  <p><%= @recipe.stuff %></p>
  <h3>Calorias</h3>
  <p><%= @recipe.calories %></p>
  <h3>Modo de preparo</h3>
  <p><%= @recipe.prepare_mode %></p>
  <h3>Custo</h3>
  <p><%= number_to_currency(@recipe.cost) %></p>
</article>

<%= link_to "Receitas", recipes_path %>
<%= link_to "Editar Receita", edit_recipe_path(@recipe) %>
<%= link_to "Deletar Receita", recipe_path(@recipe), method: :delete, data: { confirm: "Tem certeza que deseja excluir esta receita?" } %>
```





Calorias

280

Modo de preparo

Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinhe frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.

Custo

\$50.00

[Receitas](#) [Editar Receita](#) [Deletar Receita](#)

4 Receitas

Strogonoff de carne

Ingredientes: 1 kg de carne, 3 cebolas médias, sal...
Calorias: 150
Modo de preparo: Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne e deixe cozinhar por 20 minutos. Acrescente o ketchup e deixe cozinhar até obter um molho consistente e cremoso. Desligue o fogo e acrescente o creme de leite sem soro. Mexa até incorporar o molho ao creme. Coloque em uma forma refratária e decore com tempero e batata palha.
Custo: \$45.00

Filé de frango à parmegiana

Ingredientes: 1/2 kg de filé de frango, 2 ovos...
Calorias: 160
Modo de preparo: Tempere os filetes de frango com o alho, suco de limão, ervas finas e sal à gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo não muito quente. Retire e deixe descansar. Coloque os filetes em um refratário. Bubra com a mussarela em fatias. Regue com o molho e leve ao forno bem quente, somente para derreter a mussarela. Retire e sirva.
Custo: \$50.00

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite
Calorias: 110
Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga
Custo: \$60.00

Salada de peito de peru

Ingredientes: 1 maço de alface (ou rúcula), suco de...
Calorias: Light (menos de 100 calorias)
Modo de preparo: Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o açete e o sal e tempere a salada
Custo: \$20.00

[Nova receita](#)

ADICIONANDO AS ALTERAÇÕES NO GIT

```
git add .
```

```
git status
```

```
git commit -m "Deletando receitas"
```

```
C:\Sites\receitasnet>git add .
warning: LF will be replaced by CRLF in app/controllers/recipes_controller.rb.
The file will have its original line endings in your working directory

C:\Sites\receitasnet>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   app/controllers/recipes_controller.rb
    modified:   app/views/recipes/show.html.erb

C:\Sites\receitasnet>git commit -m "Deletando receitas"
[master be1d20f] Deletando receitas
 2 files changed, 7 insertions(+)
```

```
git push heroku master
```

```
heroku open
```

The screenshot shows a web browser window with the title 'Receitasnet'. The URL in the address bar is <https://receitasnet-robertopinheiro.herokuapp.com/recipes>. The page content is as follows:

3 Receitas

Salada de peito de peru

Ingredientes: 1 maço de alface(ou rúcula), suco de...

Calorias: Light (menos de 100 calorias)

Modo de preparo: Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão . Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada

Custo: \$20.00

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 115

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo: \$60.00

Fettuccine

Ingredientes: 250g de Fettuccine, 70g bacon, 2...

Calorias: 280

Modo de preparo: Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve. na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.

Custo: \$50.00

[Nova receita](#)

A screenshot of a Microsoft Edge browser window. The title bar says "Receitasnet". The address bar shows the URL "https://receitasnet-robertopinheiro.herokuapp.com/recipes/3". The main content area displays a recipe for "Fettuccine".

Fettuccine

Ingredientes

250g de Fettuccine, 70g bacon, 2 caixas de creme de leite, 3 colheres (sopa) de queijo ralado de saquinho (parmesão), 2 dentes de alho, 1 colher (de sopa) de manteiga/margarina, sal a gosto.

Calorias

280

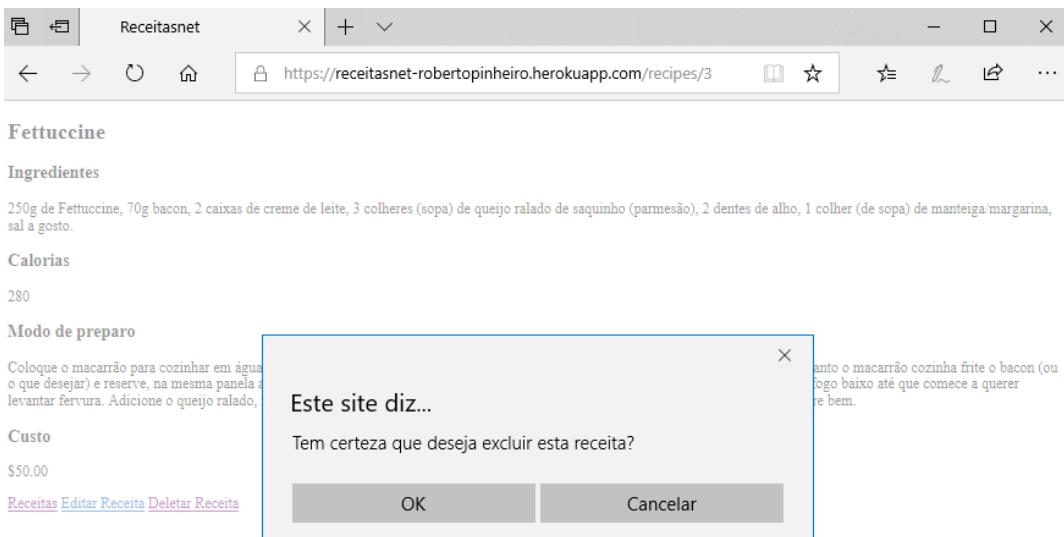
Modo de preparo

Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco do queijo. Adicione o fettuccine e o bacon e misture bem.

Custo

\$50.00

[Receitas](#) [Editar Receita](#) [Deletar Receita](#)

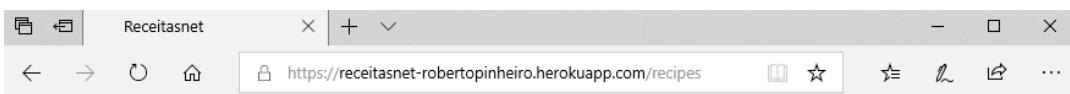
A screenshot of a Microsoft Edge browser window. The title bar says "Receitasnet". The address bar shows the URL "https://receitasnet-robertopinheiro.herokuapp.com/recipes/3". A confirmation dialog box is displayed in the center of the screen.

Este site diz...

Tem certeza que deseja excluir esta receita?

OK **Cancelar**

The background page shows the same Fettuccine recipe as the previous screenshot, with the addition of a note in the "Modo de preparo" section: "Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco do queijo. Adicione o fettuccine e o bacon e misture bem."

A screenshot of a Microsoft Edge browser window. The title bar says "Receitasnet". The address bar shows the URL "https://receitasnet-robertopinheiro.herokuapp.com/recipes".

2 Receitas

Salada de peito de peru

Ingredientes: 1 maço de alface (ou rúcula), suco de...

Calorias: Light (menos de 100 calorias)

Modo de preparo: Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada

Custo: \$20.00

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 115

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo: \$60.00

[Nova receita](#)

AULA 14 – PARTIALS

Partials são usados para eliminar repetições de código. Um arquivo partial começa sempre com underline (exemplo: _form.html.erb)

`app/views/recipes/_form.html.erb`

```
<%= form_for(@recipe) do |f| %>
  <p>
    <%= f.label :Nome %>
    <%= f.text_field :name %>
  </p>
  <p class="form-group">
    <%= f.label :Ingredientes %>
    <%= f.text_area :stuff, cols: 20, rows: 5 %>
  </p>
  <p class="form-group">
    <%= f.label :Calorias %>
    <%= f.number_field :calories %>
  </p>
  <p class="form-group">
    <%= f.label :Modo_de_Preparo %>
    <%= f.text_area :prepare_mode, cols: 20, rows: 5 %>
  </p>
  <p class="form-group">
    <%= f.label :Custo %>
    <%= f.number_field :cost %>
  </p>
  <p class="form-group">
    <%= f.submit "Salvar receita" %>
    <%= link_to "Cancelar alterações", root_path %>
  </p>
<% end %>
```

app/views/recipes/edit.html.erb

```
<h1>Editando receitas</h1>  
<%= render 'form' %>
```

app/views/recipes/new.html.erb

```
<h1>Criando receitas</h1>  
<%= render 'form' %>
```

← → C ⓘ localhost:3000/recipes/1/edit

Editando receitas

Nome

Strogonoff de carne

Ingredientes

1 kg de carne, 3 cebolas médias, sal e
pimenta-do-reino à gosto, 2 latas de
creme de leite, 2 colheres (de sopa) de
ketchup

Calorias

100

Modo de preparo

Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne e deixe cozinhar por 20 minutos. Acrescente o ketchup e deixe cozinhar até obter um molho

Custo

45

[Cancelar alterações](#)

Criando receitas

Nome

Ingredientes

Calorias

Modo de preparo

Custo

[Salvar receita](#)

[Cancelar alterações](#)

Criando receitas

Nome

Ingredientes

Calorias

Modo de preparo

...

Custo

[Salvar receita](#)

[Cancelar alterações](#)

Receitasnet x +

← → ⟳ ↑

localhost:3000/recipes/7 67% ... ⋮

Fettuccine

Ingredientes
 250g de Fettuccine, 70g bacon, 2 caixas de creme de leite, 3 colheres (sopa) de queijo ralado de saquinho (parmesão), 2 dentes de alho, 1 colher (de sopa) de manteiga/margarina, sal a gosto.

Calorias
 280

Modo de preparo
 Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junta o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.

Custo
 \$50.00

[Receitas](#) [Editar Receita](#) [Deletar Receita](#)

app/controllers/recipes_controller.rb

```
class RecipesController < ApplicationController

  def index
    @recipes = Recipe.all
  end

  def show
    @recipe = Recipe.find(params[:id])
  end

  def new
    @recipe = Recipe.new
  end

  def create
    @recipe = Recipe.new(recipe_params)
    @recipe.save
    redirect_to @recipe
  end

  def edit
    @recipe = Recipe.find(params[:id])
  end

  def update
    @recipe = Recipe.find(params[:id])
    @recipe.update(recipe_params)
    redirect_to @recipe
  end

  def destroy
    @recipe = Recipe.find(params[:id])
    @recipe.destroy
    redirect_to recipes_url
  end

  private

  def recipe_params
    params.require(:recipe).permit(:name, :stuff, :calories, :prepare_mode, :cost)
  end
end
```

ADICIONANDO AS ALTERAÇÕES NO GIT

```
git add .
```

```
git status
```

```
git commit -m "Usando partials"
```

```
C:\Sites\receitasnet>git add .
warning: LF will be replaced by CRLF in app/controllers/recipes_controller.rb.
The file will have its original line endings in your working directory

C:\Sites\receitasnet>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   app/controllers/recipes_controller.rb
    new file:   app/views/recipes/_form.html.erb
    modified:   app/views/recipes/edit.html.erb
    modified:   app/views/recipes/new.html.erb

C:\Sites\receitasnet>git commit -m "Usando partials"
[master 99fa737] Usando partials
 4 files changed, 13 insertions(+), 54 deletions(-)
 copy app/views/recipes/{edit.html.erb => _form.html.erb} (100%)
 rewrite app/views/recipes/edit.html.erb (100%)
 rewrite app/views/recipes/new.html.erb (100%)
```

```
git push heroku master
```

```
heroku open
```

The screenshot shows a web browser window with the URL <https://receitasnet-robertopinheiro.herokuapp.com/recipes/new>. The page title is "Criando receitas". The form fields are:

- Nome: An input field.
- Ingredientes: A large text area.
- Calorias: An input field.
- Modo de preparo: A large text area.
- Custo: An input field.
- Buttons: "Salvar receita" and "Cancelar alterações".

AULA 15 – FILTROS CUSTOMIZADOS

rails c

Contando e listando todas as receitas

Recipe

```
C:\Sites\receitasnet>rails c
Loading development environment (Rails 5.2.3)
irb(main):001:0> Recipe
=> Recipe (call 'Recipe.connection' to establish a connection)
irb(main):002:0>
```

Recipe.count

```
irb(main):002:0> Recipe.count
  (0.6ms)  SET NAMES utf8,  @@SESSION.sql_mode = CONCAT(CONCAT(@@sql_mode, ',STRICT_ALL_TABLES'), ',',
NO_AUTO_VALUE_ON_ZERO'),  @@SESSION.sql_auto_is_null = 0, @@SESSION.wait_timeout = 2147483
  (1.0ms)  SELECT COUNT(*) FROM `recipes`
=> 5
```

Recipe.all

```
irb(main):003:0> Recipe.all
  Recipe Load (1.2ms)  SELECT `recipes`.* FROM `recipes` LIMIT 11
=> #<ActiveRecord::Relation [#<Recipe id: 1, name: "Strogonoff de carne", stuff: "1 kg de carne, 3 ce
bolas médias, sal e pimenta-do...", calories: 150, created_at: "2019-10-09 00:45:45", updated_at: "2
019-10-10 15:01:03", prepare_mode: "Em uma panela, adicione o óleo, a carne, a cebola,...", cost: 45>
, #<Recipe id: 2, name: "Filé de frango à parmegiana", stuff: "1/2 kg de filé de frango, 2 ovos batid
os com 1 pit...", calories: 160, created_at: "2019-10-09 01:23:38", updated_at: "2019-10-09 15:32:48"
, prepare_mode: "Tempere os filetes de frango com o alho, suco de lim...", cost: 50>, #<Recipe id: 4, n
ame: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite ", calories: 110, created_at: "2019-10-
09 02:34:04", updated_at: "2019-10-09 15:42:17", prepare_mode: "Retire as espinhas do filé, passe o s
al grosso e a...", cost: 60>, #<Recipe id: 5, name: "Salada de peito de peru", stuff: "1 maço de alfa
ce(ou rúcula), suco de limão, 200g d...", calories: 95, created_at: "2019-10-09 20:38:50", updated_at
: "2019-10-09 20:38:50", prepare_mode: "Lave a alface (ou rúcula) e arrume em uma travessa...", cost:
20>, #<Recipe id: 7, name: "Fettuccine", stuff: "250g de Fettuccine, 70g bacon, 2 caixas de creme d.
...", calories: 280, created_at: "2019-10-10 22:03:37", updated_at: "2019-10-10 22:03:37", prepare_mod
e: "Coloque o macarrão para cozinhar em água salgada, ...", cost: 50>]>
```

ORDENANDO

Em ordem crescente

```
Recipe.order('created_at')
```

```
irb(main):004:0> Recipe.order('created_at')
  Recipe Load (1.1ms)  SELECT `recipes`.* FROM `recipes` ORDER BY created_at LIMIT 11
=> #<ActiveRecord::Relation [<Recipe id: 1, name: "Strogonoff de carne", stuff: "1 kg de carne, 3 cebolas médias, sal e pimenta-do...", calories: 150, created_at: "2019-10-09 00:45:45", updated_at: "2019-10-10 15:01:03", prepare_mode: "Em uma panela, adicione o óleo, a carne, a cebola,...", cost: 45>, <Recipe id: 2, name: "Fileé de frango á parmegiana", stuff: "1/2 kg de fileé de frango, 2 ovos batidos com 1 pit...", calories: 160, created_at: "2019-10-09 01:23:38", updated_at: "2019-10-09 15:32:48", prepare_mode: "Tempere os fileés de frango com o alho, suco de lim...", cost: 50>, <Recipe id: 4, name: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite", calories: 110, created_at: "2019-10-09 02:34:04", updated_at: "2019-10-09 15:42:17", prepare_mode: "Retire as espinhas do fileé, passe o sal grosso e a...", cost: 60>, <Recipe id: 5, name: "Salada de peito de peru", stuff: "1 maço de alface(ou rúcula), suco de limão, 200g d...", calories: 95, created_at: "2019-10-09 20:38:50", updated_at: "2019-10-09 20:38:50", prepare_mode: "Lave a alface (ou rúcula) e arrume em uma travessa...", cost: 20>, <Recipe id: 7, name: "Fettuccine", stuff: "250g de Fettuccine, 70g bacon, 2 caixas de creme d...", calories: 280, created_at: "2019-10-10 22:03:37", updated_at: "2019-10-10 22:03:37", prepare_mode: "Coloque o macarrão para cozinhar em água salgada, ...", cost: 50>]>
```

Em ordem decrescente

```
Recipe.order('created_at desc')
```

```
irb(main):006:0> Recipe.order('created_at desc')
  Recipe Load (62.8ms)  SELECT `recipes`.* FROM `recipes` ORDER BY created_at desc LIMIT 11
=> #<ActiveRecord::Relation [<Recipe id: 7, name: "Fettuccine", stuff: "250g de Fettuccine, 70g bacon, 2 caixas de creme d...", calories: 280, created_at: "2019-10-10 22:03:37", updated_at: "2019-10-10 22:03:37", prepare_mode: "Coloque o macarrão para cozinhar em água salgada, ...", cost: 50>, <Recipe id: 5, name: "Salada de peito de peru", stuff: "1 maço de alface(ou rúcula), suco de limão, 200g d...", calories: 95, created_at: "2019-10-09 20:38:50", updated_at: "2019-10-09 20:38:50", prepare_mode: "Lave a alface (ou rúcula) e arrume em uma travessa...", cost: 20>, <Recipe id: 4, name: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite", calories: 110, created_at: "2019-10-09 02:34:04", updated_at: "2019-10-09 15:42:17", prepare_mode: "Retire as espinhas do fileé, passe o sal grosso e a...", cost: 60>, <Recipe id: 2, name: "Fileé de frango á parmegiana", stuff: "1/2 kg de fileé de frango, 2 ovos batidos com 1 pit...", calories: 160, created_at: "2019-10-09 01:23:38", updated_at: "2019-10-09 15:32:48", prepare_mode: "Tempere os fileés de frango com o alho, suco de lim...", cost: 50>, <Recipe id: 1, name: "Strogonoff de carne", stuff: "1 kg de carne, 3 cebolas médias, sal e pimenta-do...", calories: 150, created_at: "2019-10-09 00:45:45", updated_at: "2019-10-10 15:01:03", prepare_mode: "Em uma panela, adicione o óleo, a carne, a cebola,...", cost: 45>]>
```

BUSCAS

Buscando um receita por seu id

```
Recipe.find(1)
```

```
irb(main):007:0> Recipe.find(1)
  Recipe Load (2.6ms)  SELECT `recipes`.* FROM `recipes` WHERE `recipes`.`id` = 1 LIMIT 1
=> #<Recipe id: 1, name: "Strogonoff de carne", stuff: "1 kg de carne, 3 cebolas médias, sal e pimenta-do...", calories: 150, created_at: "2019-10-09 00:45:45", updated_at: "2019-10-10 15:01:03", prepare_mode: "Em uma panela, adicione o óleo, a carne, a cebola,...", cost: 45>
```

Buscando uma receita pelo nome

```
Recipe.find_by(name: "Salmão ao forno")
```

```
irb(main):009:0> Recipe.find_by(name: "Salmão ao forno")
  Recipe Load (1.7ms)  SELECT `recipes`.* FROM `recipes` WHERE `recipes`.`name` = 'Salmão ao forno'
LIMIT 1
=> #<Recipe id: 4, name: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite ", calories: 110, created_at: "2019-10-09 02:34:04", updated_at: "2019-10-09 15:42:17", prepare_mode: "Retire as espinhas do filé, passe o sal grosso e a...", cost: 60>
```

Buscando receitas com calorias menores que 100 (lighths)

```
Recipe.where('calories < 100')
```

```
irb(main):010:0> Recipe.where('calories < 100')
  Recipe Load (69.6ms)  SELECT `recipes`.* FROM `recipes` WHERE (calories < 100) LIMIT 11
=> #<ActiveRecord::Relation [<Recipe id: 5, name: "Salada de peito de peru", stuff: "1 maço de alface(ou rúcula), suco de limão, 200g d...", calories: 95, created_at: "2019-10-09 20:38:50", updated_at: "2019-10-09 20:38:50", prepare_mode: "Lave a alface (ou rúcula) e arrume em uma travessa...", cost: 20>]>
```

```
Recipe.where('calories < ?', 100)
```

```
irb(main):011:0> Recipe.where('calories < ?', 100)
  Recipe Load (0.5ms)  SELECT `recipes`.* FROM `recipes` WHERE (calories < 100) LIMIT 11
=> #<ActiveRecord::Relation [<Recipe id: 5, name: "Salada de peito de peru", stuff: "1 maço de alface(ou rúcula), suco de limão, 200g d...", calories: 95, created_at: "2019-10-09 20:38:50", updated_at: "2019-10-09 20:38:50", prepare_mode: "Lave a alface (ou rúcula) e arrume em uma travessa...", cost: 20>]>
```

Busca usando um método do rails

```
Recipe.where('created_at < ?', Time.now)
```

```
irb(main):012:0> Recipe.where('created_at < ?', Time.now)
  Recipe Load (64.7ms)  SELECT `recipes`.* FROM `recipes` WHERE (created_at < '2019-10-10 22:25:45.236308') LIMIT 11
=> #<ActiveRecord::Relation [<Recipe id: 1, name: "Strogonoff de carne", stuff: "1 kg de carne, 3 cebolas médias, sal e pimenta-do...", calories: 150, created_at: "2019-10-09 00:45:45", updated_at: "2019-10-10 15:01:03", prepare_mode: "Em uma panela, adicione o óleo, a carne, a cebola,...", cost: 45>, <Recipe id: 2, name: "Filé de frango à parmegiana", stuff: "1/2 kg de filé de frango, 2 ovos batidos com 1 pit...", calories: 160, created_at: "2019-10-09 01:23:38", updated_at: "2019-10-09 15:32:48", prepare_mode: "Tempere os filés de frango com o alho, suco de lim...", cost: 50>, <Recipe id: 4, name: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite ", calories: 110, created_at: "2019-10-09 02:34:04", updated_at: "2019-10-09 15:42:17", prepare_mode: "Retire as espinhas do filé, passe o sal grosso e a...", cost: 60>, <Recipe id: 5, name: "Salada de peito de peru", stuff: "1 maço de alface(ou rúcula), suco de limão, 200g d...", calories: 95, created_at: "2019-10-09 20:38:50", updated_at: "2019-10-09 20:38:50", prepare_mode: "Lave a alface (ou rúcula) e arrume em uma travessa...", cost: 20>, <Recipe id: 7, name: "Fettuccine", stuff: "250g de Fettuccine, 70g bacon, 2 caixas de creme d...", calories: 280, created_at: "2019-10-10 22:03:37", updated_at: "2019-10-10 22:03:37", prepare_mode: "Coloque o macarrão para cozinhar em água salgada, ...", cost: 50>]>
```

Busca encadeando métodos

```
Recipe.where('calories > 100').order('created_at desc')
```

```
irb(main):013:0> Recipe.where('calories > 100').order('created_at desc')
   Recipe Load (7.7ms)  SELECT `recipes`.* FROM `recipes` WHERE (calories > 100) ORDER BY created_at
desc LIMIT 11
=> #<ActiveRecord::Relation [#<Recipe id: 7, name: "Fettuccine", stuff: "250g de Fettuccine, 70g bacon, 2 caixas de creme d...", calories: 280, created_at: "2019-10-10 22:03:37", updated_at: "2019-10-10 22:03:37", prepare_mode: "Coloque o macarrão para cozinhar em água salgada, ...", cost: 50>, #<Recipe id: 4, name: "Salmão ao forno", stuff: "1 salmão, sal grosso e azeite", calories: 110, created_at: "2019-10-09 02:34:04", updated_at: "2019-10-09 15:42:17", prepare_mode: "Retire as espinhas do filé, passe o sal grosso e a...", cost: 60>, #<Recipe id: 2, name: "Filé de frango à parmegiana", stuff: "1/2 kg de filé de frango, 2 ovos batidos com 1 pit...", calories: 160, created_at: "2019-10-09 01:23:38", updated_at: "2019-10-09 15:32:48", prepare_mode: "Tempere os filés de frango com o alho, suco de lim...", cost: 50>, #<Recipe id: 1, name: "Strogonoff de carne", stuff: "1 kg de carne, 3 cebolas médias, sal e pimenta-do...", calories: 150, created_at: "2019-10-09 00:45:45", updated_at: "2019-10-09 15:01:03", prepare_mode: "Em uma panela, adicione o óleo, a carne, a cebola,...", cost: 45>]>
```

- Regra de negócios criar dentro do modelo.

app/models/recipe.rb

```
class Recipe < ApplicationRecord
  def light?
    calories < 100
  end

  def self.created_at
    where('created_at <= ?', Time.now).order('created_at desc')
  end
end
```

app/controllers/recipes_controller.rb

```
class RecipesController < ApplicationController

  def index
    @recipes = Recipe.created_at
  end

  def show
    @recipe = Recipe.find(params[:id])
  end

  def new
    @recipe = Recipe.new
  end

  def create
    @recipe = Recipe.new(recipe_params)
    @recipe.save
    redirect_to @recipe
  end

  def edit
    @recipe = Recipe.find(params[:id])
  end

  def update
    @recipe = Recipe.find(params[:id])
    @recipe.update(recipe_params)
    redirect_to @recipe
  end

  def destroy
    @recipe = Recipe.find(params[:id])
    @recipe.destroy
    redirect_to recipes_url
  end

  private

  def recipe_params
    params.require(:recipe).permit(:name, :stuff, :calories, :prepare_mode, :cost)
  end
end
```

4 Receitas

Salada de peito de peru

Ingredientes: 1 maço de alface(ou rúcula), suco de...

Calorias: Light (menos de 100 calorias)

Modo de preparo: Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada

Custo: \$20.00

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 110

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo: \$60.00

File de frango à parmegiana

Ingredientes: 1/2 kg de filé de frango, 2 ovos...

Calorias: 160

Modo de preparo: Tempere os filetes de frango com o alho, suco de limão, ervas finas e sal à gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo não muito quente. Retire e deixe descansar. Coloque os filetes em um refratário. Bubra com a mussarela em fatias. Regue com o molho e leve ao forno bem quente, somente para derreter a mussarela. Retire e sirva.

Custo: \$50.00

Strogonoff de carne

Ingredientes: 1 kg de carne, 3 cebolas médias, sal...

Calorias: 100

Modo de preparo: Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne e deixe cozinhar por 20 minutos. Acrescente o ketchup e deixe cozinhar até obter um molho consistente e cremoso. Desligue o fogo e acrescente o creme de leite sem soro. Mexa até incorporar o molho ao creme. Coloque em uma forma refratária e decore com tempero e batata palha.

Custo: \$45.00

[Nova receita](#)

ADICIONANDO AS ALTERAÇÕES NO GIT

```
git add .
git status
git commit -m "Criando filtros customizados"
```

```
C:\Sites\receitasnet>git add .
warning: LF will be replaced by CRLF in app/controllers/recipes_controller.rb.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in app/models/recipe.rb.
The file will have its original line endings in your working directory

C:\Sites\receitasnet>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   app/controllers/recipes_controller.rb
    modified:   app/models/recipe.rb

C:\Sites\receitasnet>git commit -m "Criando filtros customizados"
[master 114f455] Criando filtros customizados
 2 files changed, 5 insertions(+), 1 deletion(-)
```

```
git push heroku master
```

```
heroku open
```



2 Receitas

Salada de peito de peru

Ingredientes: 1 maço de alface(ou rúcula), suco de...

Calorias: Light (menos de 100 calorias)

Modo de preparo: Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada

Custo: \$20.00

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 115

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo: \$60.00

[Nova receita](#)



Criando receitas

Nome

Ingredientes

```
250g de Fettuccine, 70g bacon, 2 caixas  
de creme de leite, 3 colheres (sopa) de  
queijo ralado de saquinho (parmesão), 2  
dentes de alho, 1 colher (de sopa) de  
manteiga/margarina, sal a gosto.
```

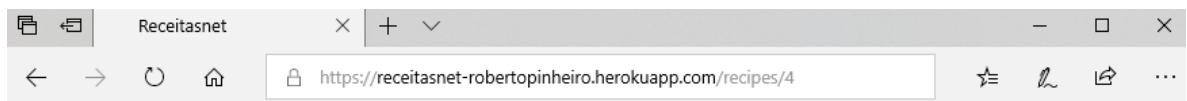
Calorias

Modo de preparo

```
até que comece a querer levantar  
fervura. Adicione o queijo ralado,  
misture rapidamente para derreter um  
pouco o queijo. Adicione o fettuccine  
e o bacon e misture bem.
```

Custo

[Cancelar alterações](#)



Fettuccine

Ingredientes

250g de Fettuccine, 70g bacon, 2 caixas de creme de leite, 3 colheres (sopa) de queijo ralado de saquinho (parmesão), 2 dentes de alho, 1 colher (de sopa) de manteiga/margarina, sal a gosto.

Calorias

280

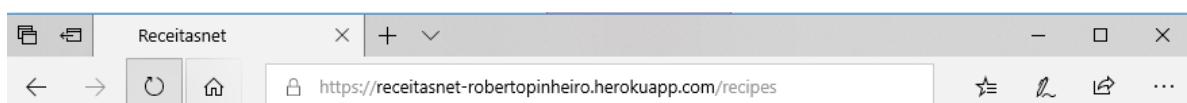
Modo de preparo

Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco do queijo. Adicione o fettuccine e o bacon e misture bem.

Custo

\$50.00

[Receitas](#) [Editar Receita](#) [Deletar Receita](#)



3 Receitas

Fettuccine

Ingredientes: 250g de Fettuccine, 70g bacon, 2...

Calorias: 280

Modo de preparo: Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco do queijo. Adicione o fettuccine e o bacon e misture bem.

Custo: \$50.00

Salada de peito de peru

Ingredientes: 1 maço de alface(ou rúcula), suco de...

Calorias: Light (menos de 100 calorias)

Modo de preparo: Lave a alface (ou rúcula) e arrume em uma travessa. Regue com o suco de limão. Alterne o peito de peru e o alface. Coloque o cream cheese no centro do prato. Misture o azeite, o aceto e o sal e tempere a salada

Custo: \$20.00

Salmão ao forno

Ingredientes: 1 salmão, sal grosso e azeite

Calorias: 115

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe cozinhar por 1h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis com cenourinha picada e cozida, refogada na manteiga

Custo: \$60.00

[Nova receita](#)

AULA 16 – NOVAS MIGRAÇÕES

- Incluir novos campos, inclusive para imagens.

CRIANDO A MIGRATION

```
rails g migration AddFieldsToRecipe kind:string portion:string duration:datetime poster:string
```

```
C:\Sites\receitasnet>rails g migration AddFieldsToRecipe kind:string portion:string duration:datetime poster:string
      invoke  active_record
      create    db/migrate/20191011014729_add_fields_to_recipe.rb
```

db/migrate/20191011014729_add_fields_to_recipe.rb

```
class AddFieldsToRecipe < ActiveRecord::Migration[5.2]
  def change
    add_column :recipes, :kind, :string
    add_column :recipes, :portion, :string
    add_column :recipes, :duration, :datetime
    add_column :recipes, :poster, :string
  end
end
```

- Execute a migration

```
rake db:migrate
```

```
C:\Sites\receitasnet>rake db:migrate
== 20191011014729 AddFieldsToRecipe: migrating =====
-- add_column(:recipes, :kind, :string)
  -> 1.5593s
-- add_column(:recipes, :portion, :string)
  -> 0.6630s
-- add_column(:recipes, :duration, :datetime)
  -> 0.6694s
-- add_column(:recipes, :poster, :string)
  -> 0.5416s
== 20191011014729 AddFieldsToRecipe: migrated (3.4356s) =====
```

Colunas:		Adicionar	Remover	Mover para cima	Mover para baixo		
#	Nome	Tipo de dados	Tamanho/Itens	Unsign...	Permitir...	Zerofill	Padrão
1	id	BIGINT	20	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	name	VARCHAR	255	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	NULL
3	stuff	TEXT		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	NULL
4	calories	INT	11	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
5	created_at	DATETIME		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Nenhum padrão
6	updated_at	DATETIME		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Nenhum padrão
7	prepare_mode	TEXT		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	NULL
8	cost	DECIMAL	10,0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
9	kind	VARCHAR	255	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	NULL
10	portion	VARCHAR	255	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	NULL
11	duration	DATETIME		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	NULL
12	poster	VARCHAR	255	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	NULL

Desfazendo uma migration

- Para desfazer uma migration use o comando:

rake db:rollback

```
receitasnet (master) $ rake db:rollback
== 20160509163111 AddFieldsToRecipe: reverting =====
-- remove_column(:recipes, :poster, :string)
  -> 0.0162s
-- remove_column(:recipes, :duration, :datetime)
  -> 0.0089s
-- remove_column(:recipes, :portion, :string)
  -> 0.0116s
-- remove_column(:recipes, :kind, :string)
  -> 0.0144s
== 20160509163111 AddFieldsToRecipe: reverted (0.0552s) =====
```

```

receitasnet (master) $ rake db:migrate:status
database: /Users/fabriciomattos/Desktop/codifike/receitasnet/db/development.sqlite3

  Status  Migration ID      Migration Name
  ----
    up    20160503021434  Create recipes
    up    20160503181940  Add new fields to recipe
  down   20160509163111  Add fields to recipe

receitasnet (master) $ rake db:migrate
== 20160509163111 AddFieldsToRecipe: migrating =====
-- add_column(:recipes, :kind, :string)
  -> 0.0028s
-- add_column(:recipes, :portion, :string)
  -> 0.0006s
-- add_column(:recipes, :duration, :datetime)
  -> 0.0004s
-- add_column(:recipes, :poster, :string)
  -> 0.0004s
== 20160509163111 AddFieldsToRecipe: migrated (0.0047s) =====

receitasnet (master) $ rake db:migrate:status
database: /Users/fabriciomattos/Desktop/codifike/receitasnet/db/development.sqlite3

  Status  Migration ID      Migration Name
  ----
    up    20160503021434  Create recipes
    up    20160503181940  Add new fields to recipe
    up    20160509163111  Add fields to recipe

```

app/models/recipe.rb

```

class Recipe < ActiveRecord::Base
  def light?
    calories < 100
  end

  KIND = %w(Carnes Aves Peixes Massas Saladas Doces)

  def self.created_at
    where('created_at <= ?', Time.now).order('created_at desc')
  end
end

```

app/views/recipes/_form.html.erb

```
<%= form_for(@recipe) do |f| %>

  <p>
    <%= f.label :Nome %><br>
    <%= f.text_field :name %>
  </p>

  <p>
    <%= f.label :Ingredientes %><br>
    <%= f.text_area :stuff, cols: 20, rows: 5 %>
  </p>

  <p>
    <%= f.label :Tipo %><br>
    <%= f.select :kind, Recipe::KIND, prompt: 'Escolha uma opção' %>
  </p>

  <p>
    <%= f.label :Calorias %><br>
    <%= f.number_field :calories %>
  </p>

  <p>
    <%= f.label :Modo_de_Preparo %><br>
    <%= f.text_area :prepare_mode, cols: 20, rows: 5 %>
  </p>

  <p>
    <%= f.label :Tempo_de_Preparo %><br>
    <%= f.time_select :duration %>
  </p>

  <p>
    <%= f.label :Porções %><br>
    <%= f.text_field :portion %>
  </p>

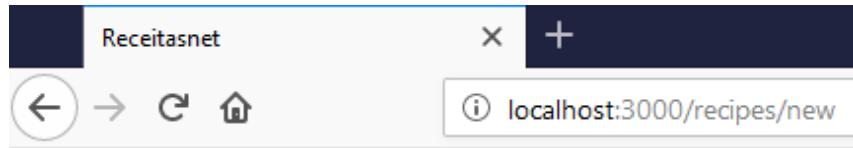
  <p>
    <%= f.label :Custo %><br>
    <%= f.number_field :cost %>
  </p>
```

```

<p>
  <%= f.label :Poster %><br>
  <%= f.text_field :poster, placeholder: ".gif, .jpeg, .png" %>
</p>

<p>
  <%= f.submit "Salvar receita", class:"btn btn-outline-success" %>
  <%= link_to "Cancelar alterações", root_path, class:"btn btn-outline-danger" %>
</p>
<% end %>

```



Criando receitas

Nome

Ingredientes

Tipo

Calorias
 [▲ ▼]

Modo de preparo

Tempo de preparo
 :

Porções

Custo
 [▲ ▼]

Poster
 .gif, .jpeg, .png

[Cancelar alterações](#)

app/controllers/recipes_controller.rb

```
class RecipesController < ApplicationController

  def index
    @recipes = Recipe.created_at
  end

  def show
    @recipe = Recipe.find(params[:id])
  end

  def new
    @recipe = Recipe.new
  end

  def create
    @recipe = Recipe.new(recipe_params)
    @recipe.save
    redirect_to @recipe
  end

  def edit
    @recipe = Recipe.find(params[:id])
  end

  def update
    @recipe = Recipe.find(params[:id])
    @recipe.update(recipe_params)
    redirect_to @recipe
  end

  def destroy
    @recipe = Recipe.find(params[:id])
    @recipe.destroy
    redirect_to recipes_url
  end

  private

  def recipe_params
    params.require(:recipe).permit(:name, :stuff, :calories, :prepare_mode, :cost, :kind, :portion, :duration, :poster)
  end

end
```

app/views/recipes/index.html.erb

```
<h1><%= pluralize(@recipes.size, "Receita", "Receitas") %></h1>
<% @recipes.each do |recipe| %>
<article>
<header>
  <h2><%= link_to recipe.name, recipe %></h2>
  <%= imagem(recipe) %>
</header>
<p><strong>Ingredientes: </strong><%= truncate(recipe.stuff, length:40, separator:' ') %></p>
<p><strong>Calorias: </strong><%= formata_caloria(recipe) %></p>
<p><strong>Tempo de preparo: </strong><%= recipe.duration.to_formatted_s(:time) %> min</p>
<p><strong>Modo de preparo: </strong><%= recipe.prepare_mode %></strong></p>
<p><strong>Custo: </strong><%= number_to_currency(recipe.cost) %></strong></p>
</article>
<% end %>

<br>

<%= link_to 'Nova receita', new_recipe_path %>
```

app/views/recipes/show.html.erb

```
<article>
<head>
  <h1><%= @recipe.name %></h1>
  <%= imagem(@recipe) %>
</head>
<h3>Tipo</h3>
<p><%= @recipe.kind %></p>

<h3>Ingredientes</h3>
<p><%= @recipe.stuff %></p>

<h3>Calorias</h3>
<p><%= @recipe.calories %></p>

<h3>Modo de preparo</h3>
<p><%= @recipe.prepare_mode %></p>

<h3>Tempo de preparo</h3>
<p><%= @recipe.duration.to_formatted_s(:time) %> min</p>
```

```
<h3>Porções</h3>
<p><%= pluralize(@recipe.portion, 'porção', 'porções') %></p>

<h3>Custo</h3>
<p><%= number_to_currency(@recipe.cost) %></p>
</article>

<%= link_to "Receitas", recipes_path %>
<%= link_to "Editar Receita", edit_recipe_path(@recipe) %>
<%= link_to "Deletar Receita", recipe_path(@recipe), method: :delete, data: { confirm: "Tem certeza que deseja excluir esta receita?" } %>
```

app/helpers/recipes_helper.rb

```
module RecipesHelper
  def formata_caloria(recipe)
    if recipe.light?
      content_tag(:strong, "Light (menos de 100 calorias)")
    else
      recipe.calories
    end
  end

  def imagem(recipe)
    if recipe.poster.blank?
      image_tag('no-photo.png')
    else
      image_tag(recipe.poster)
    end
  end
end
```

SEEDS

Permite popular as tabelas com dados pré-estabelecidos.

db/seeds.rb

```
Recipe.create!([
  {
    name: "Estrogonofe de carne",
    stuff: "1 kg de carne,3 cebolas médias, sal e pimenta-do-reino a gosto, 2 latas de creme de leite, 2 colheres (sopa) de ketchup",
    calories: 110,
    kind: "Carnes",
    prepare_mode: "Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne. Acrescente o ketchup e o champignon. Desligue o fogo e acrescente o creme de leite sem soro. Coloque em uma forma refratária e decore com tempero e batata palha. ",
    portion: "5",
    duration: "2016-05-06 00:40:00",
    cost:50.00,
    poster: "estrogonofe-de-carne.jpg"
  },
  {
    name: "Filé de Frango a parmegiana",
    stuff: "1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 xícaras (chá) de farinha de rosca,1 caixinha de molho de tomate pronto, mussarela ",
    calories: 160,
    kind: "Aves",
    prepare_mode: "Tempere os filés de frango com o alho, suco de limão, ervas finas e sal a gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo. Coloque os filés em um refratório. Cubra com a mussarela. Regue com o molho e leve ao forno para derreter a mussarela. Retire e sirva ",
    portion: "8",
    duration: "2016-05-06 0:35:00",
    cost:40.00,
    poster: "file-frango-parmegiana.jpeg"
  },
  {
    name: "Salmão ao forno",
    stuff: "Sal marinho,Pimenta moída na hora, Azeite extra virgem, 2 filés de salmão 1.1kg com pele, sem escamas e espinhas, 2 limões",
    calories: 100,
    kind: "Peixes",
    prepare_mode: "Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe temperando na geladeira por 1 h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis ou com cenourinha picada e cozida, refogada na manteiga",
    portion: "4",
    duration: "2016-05-06 0:50:00",
  }
])
```

```

    cost:60.00,
    poster: "salmao-forno.jpeg"
},
{
    name: "Salada Light",
    stuff: "alface, agrião, brócolis, um mix de folhas",
    calories: 90,
    kind: "Saladas",
    prepare_mode: "Misture todas as folhas em uma tigela, tempere e sirva.",
    portion: "4",
    duration: "2016-05-06 0:10:00",
    cost:15.00,
    poster: "salada-light.jpeg"
},
{
    name: "Fettuccine",
    stuff: "250g de Fettuccine, 70g bacon, 2 caixas de creme de leite, 3 colheres (sopa) de queijo ralado de saquinho (parmesão), 2 dentes de alho, 1 colher (de sopa) de manteiga/margarina, sal a gosto.",
    calories: 280,
    kind: "Massas",
    prepare_mode: "Coloque o macarrão para cozinar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.",
    portion: "4",
    duration: "2016-05-06 0:13:00",
    cost:50.00,
    poster: ""
},
])
)

```

rake db:schema:load db:seed

```

C:\Sites\receitasnet>rake db:schema:load db:seed
-- create_table("recipes", {:options=>"ENGINE=InnoDB DEFAULT CHARSET=utf8", :force=>:cascade})
  -> 1.4110s
-- create_table("recipes", {:options=>"ENGINE=InnoDB DEFAULT CHARSET=utf8", :force=>:cascade})
  -> 0.4790s

```

db:schema:load - Apaga os dados de todas as tabelas

db:seed - Executa o seed

5 Receitas

[Estrogonofe de carne](#)



Ingredientes: 1 kg de carne, 3 cebolas médias, sal e...

Calorias: 110

Tempo de preparo: 00:40 min

Modo de preparo: Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne. Acrescente o ketchup e o champignon. Desligue o fogo e acrescente o creme de leite sem soro. Coloque em uma forma refratária e decore com tempero e banana pálha.

Custo: \$50,00

[File de Frango a parmegiana](#)



Ingredientes: 1/2 kg de filé de frango, 2 ovos...

Calorias: 160

Tempo de preparo: 00:35 min

Modo de preparo: Tempere os filetes de frango com o alho, suco de limão, ervas finas e sal a gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo. Coloque os filets em um refratário. Cubra com a mussarela. Regue com o molho e leve ao forno para derreter a mussarela. Retire e sirva.

Custo: \$40,00

[Salmão ao forno](#)



Ingredientes: Sal marinho, Pimenta moída na hora...

Calorias: 100

Tempo de preparo: 00:50 min

Modo de preparo: Reserve as espinhas do filé, pase o sal grosso e as ervas. Regue com o azeite e deixe temperando na geladeira por 1 h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis ou com cenourinha picada e coxida, refogada na manteiga.

Custo: \$60,00

[Salada Light](#)



Ingredientes: alface, agrião, brócolis, um mix de...

Calorias: Light (menos de 100 calorias)

Tempo de preparo: 00:10 min

Modo de preparo: Misture todas as folhas em uma tigela, tempere e sirva.

Custo: \$15,00

[Fettuccine](#)



IMAGEM NÃO
DISPONÍVEL

Ingredientes: 250g de Fettuccine, 70g bacon, 2...

Calorias: 280

Tempo de preparo: 00:13 min

Modo de preparo: Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.

Custo: \$50,00

[Nova receita](#)

INSERINDO UMA NOVA GEM NO ARQUIVO GEMFILE

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', '~> 5.2.1'  
# 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 'ductape'  
# 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 ActiveStorage 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 'mysql2', '~> 0.5.2'  
  gem 'web-console', '>= 3.3.0'  
end
```

```

group :production do
  gem 'pg'
  gem 'rails_12factor'
end

group :test do
  # Adds support for Capybara system testing and selenium driver
  gem 'capybara', '>= 2.15'
  gem 'selenium-webdriver'
  # Easy installation and use of chromedriver to run system tests with Chrome
  gem 'chromedriver-helper'
end

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

```

bundle install --without production

config/enviroments/production

```

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

  # Code is not reloaded between requests.
  config.cache_classes = true

  # Eager load code on boot. This eager loads most of Rails and
  # your application in memory, allowing both threaded web servers
  # and those relying on copy on write to perform better.
  # Rake tasks automatically ignore this option for performance.
  config.eager_load = true

  # Full error reports are disabled and caching is turned on.
  config.consider_all_requests_local    = false
  config.action_controller.perform_caching = true

  # Ensures that a master key has been made available in either ENV["RAILS_MASTER_KEY"]
  # or in config/master.key. This key is used to decrypt credentials (and other encrypted files).
  # config.require_master_key = true

  # Disable serving static files from the `/public` folder by default since
  # Apache or NGINX already handles this.
  # config.public_file_server.enabled = ENV['RAILS_SERVE_STATIC_FILES'].present?
  config.public_file_server.enabled = true

```

```
# Compress JavaScripts and CSS.
config.assets.js_compressor = :uglifier
# config.assets.css_compressor = :sass

# Do not fallback to assets pipeline if a precompiled asset is missed.
config.assets.compile = false
config.assets.compile = true

# `config.assets.precompile` and `config.assets.version` have moved to config/initializers/assets.rb

# Enable serving of images, stylesheets, and JavaScripts from an asset server.
# config.action_controller.asset_host = 'http://assets.example.com'

# Specifies the header that your server uses for sending files.
# config.action_dispatch.x_sendfile_header = 'X-Sendfile' # for Apache
# config.action_dispatch.x_sendfile_header = 'X-Accel-Redirect' # for NGINX

# Store uploaded files on the local file system (see config/storage.yml for options)
config.active_storage.service = :local

# Mount Action Cable outside main process or domain
# config.action_cable.mount_path = nil
# config.action_cable.url = 'wss://example.com/cable'
# config.action_cable.allowed_request_origins = [ 'http://example.com', /http://\Vexample.*/] 

# Force all access to the app over SSL, use Strict-Transport-Security, and use secure cookies.
# config.force_ssl = true

# Use the lowest log level to ensure availability of diagnostic information
# when problems arise.
config.log_level = :debug

# Prepend all log lines with the following tags.
config.log_tags = [ :request_id ]

# Use a different cache store in production.
# config.cache_store = :mem_cache_store

# Use a real queuing backend for Active Job (and separate queues per environment)
# config.active_job.queue_adapter = :resque
# config.active_job.queue_name_prefix = "receitasnet_#{Rails.env}"

config.action_mailer.perform_caching = false

# Ignore bad email addresses and do not raise email delivery errors.
# Set this to true and configure the email server for immediate delivery to raise delivery errors.
# config.action_mailer.raise_delivery_errors = false

# Enable locale fallbacks for I18n (makes lookups for any locale fall back to
# the I18n.default_locale when a translation cannot be found).
config.i18n.fallbacks = true
```

```
# Send deprecation notices to registered listeners.
config.active_support.deprecation = :notify

# Use default logging formatter so that PID and timestamp are not suppressed.
config.log_formatter = ::Logger::Formatter.new

# Use a different logger for distributed setups.
# require 'syslog/logger'
# config.logger = ActiveSupport::TaggedLogging.new(Syslog::Logger.new 'app-name')

if ENV["RAILS_LOG_TO_STDOUT"].present?
  logger      = ActiveSupport::Logger.new(STDOUT)
  logger.formatter = config.log_formatter
  config.logger  = ActiveSupport::TaggedLogging.new(logger)
end

# Do not dump schema after migrations.
config.active_record.dump_schema_after_migration = false
```

ADICIONANDO AS ALTERAÇÕES NO GIT

git status

git add .

git status

git commit -m "Criando novos campos"

git push heroku master

heroku run rake db:seed

heroku open

5 Receitas

[Fettuccine](#)



IMAGEM NÃO
DISPONÍVEL

Ingredientes: 250g de Fettuccine, 70g bacon, 2...

Calorias: 280

Tempo de preparo: 00:13 min

Modo de preparo: Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, anemone peneira adicione manteiga e refogue rapidamente o alho amassado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente, para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.

Custo: \$50,00

[Salada Light](#)



Ingredientes: alface, agrião, brócolis, um mix de...

Calorias: Light (menos de 100 calorias)

Tempo de preparo: 00:10 min

Modo de preparo: Misture todas as folhas em uma tigela, tempere e sirva.

Custo: \$15,00

[Salmão ao forno](#)



Ingredientes: Sal marinho, Pimenta moída na hora,...

Calorias: 100

Tempo de preparo: 00:50 min

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe temperando na geladeira por 1 h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis ou com cenoura picada e coada, refogada na manteiga.

Custo: \$60,00

[File de Frango a parmesana](#)



Ingredientes: 1/2 kg de file de frango, 2 ovos...

Calorias: 160

Tempo de preparo: 00:35 min

Modo de preparo: Tempere os filets de frango com o alho, suco de limão, ervas finas e sal a gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo. Coloque os filets em um refratário. Cubra com a mussarela. Regue com o molho e leve ao forno para derreter a mussarela. Retire e sirva

Custo: \$40,00

[Estragonofe de carne](#)



Ingredientes: 1 kg de carne, 3 cebolas médias, sal e...

Calorias: 110

Tempo de preparo: 00:40 min

Modo de preparo: Em uma panela, adicione óleo, a carne, a cebola, os tomates, o caldo de carne. Acrescente o ketchup e o champignon. Desligue o fogo e acrescente o creme de leite sem soro. Coloque em uma forma refratária e decore com tempero e batata palha.

Custo: \$50,00

[Nova receita](#)

AULA 17 – UM POUCO DE BELEZA COM BOOTSTRAP

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', '~> 5.2.1'
# Use Puma as the app server
gem 'puma', '~> 3.11'
gem 'bootstrap-sass'
# 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 ActiveStorage 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 'mysql2', '~> 0.5.2'
```

```
gem 'web-console', '>= 3.3.0'
end

group :production do
  gem 'pg'
  gem 'rails_12factor'
end

group :test do
  # Adds support for Capybara system testing and selenium driver
  gem 'capybara', '>= 2.15'
  gem 'selenium-webdriver'
  # Easy installation and use of chromedriver to run system tests with Chrome
  gem 'chromedriver-helper'
end

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

bundle install --without production

- Renomeie o arquivo app/assets/stylesheets/application.css para app/assets/stylesheets/application.css.scss

app/assets/stylesheets/bootstrap_receitanet.css.scss

```
@import 'bootstrap';
```

- Reinicie o servidor, para que as alterações sejam consideradas.

config/enviroments/production.rb

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

  # Code is not reloaded between requests.
  config.cache_classes = true

  # Eager load code on boot. This eager loads most of Rails and
  # your application in memory, allowing both threaded web servers
```

```
# and those relying on copy on write to perform better.
# Rake tasks automatically ignore this option for performance.
config.eager_load = true

# Full error reports are disabled and caching is turned on.
config.consider_all_requests_local    = false
config.action_controller.perform_caching = true

# Ensures that a master key has been made available in either ENV["RAILS_MASTER_KEY"]
# or in config/master.key. This key is used to decrypt credentials (and other encrypted files).
# config.require_master_key = true

# Disable serving static files from the `/public` folder by default since
# Apache or NGINX already handles this.
# config.public_file_server.enabled = ENV['RAILS_SERVE_STATIC_FILES'].present?
config.public_file_server.enabled = true

# Compress JavaScripts and CSS.
config.assets.js_compressor = :uglifier
# config.assets.css_compressor = :sass

# Do not fallback to assets pipeline if a precompiled asset is missed.
config.assets.compile = false
config.assets.compile = true

# `config.assets.precompile` and `config.assets.version` have moved to config/initializers/assets.rb

# Enable serving of images, stylesheets, and JavaScripts from an asset server.
# config.action_controller.asset_host = 'http://assets.example.com'

# Specifies the header that your server uses for sending files.
# config.action_dispatch.x_sendfile_header = 'X-Sendfile' # for Apache
# config.action_dispatch.x_sendfile_header = 'X-Accel-Redirect' # for NGINX

# Store uploaded files on the local file system (see config/storage.yml for options)
config.active_storage.service = :local

# Mount Action Cable outside main process or domain
# config.action_cable.mount_path = nil
# config.action_cable.url = 'wss://example.com/cable'
# config.action_cable.allowed_request_origins = [ 'http://example.com', /http://\V/example.*/] 

# Force all access to the app over SSL, use Strict-Transport-Security, and use secure cookies.
# config.force_ssl = true

# Use the lowest log level to ensure availability of diagnostic information
# when problems arise.
config.log_level = :debug

# Prepend all log lines with the following tags.
config.log_tags = [ :request_id ]
```

```

# Use a different cache store in production.
# config.cache_store = :mem_cache_store

# Use a real queuing backend for Active Job (and separate queues per environment)
# config.active_job.queue_adapter   = :resque
# config.active_job.queue_name_prefix = "receitasnet_#{Rails.env}"

config.action_mailer.perform_caching = false

# Ignore bad email addresses and do not raise email delivery errors.
# Set this to true and configure the email server for immediate delivery to raise delivery errors.
# config.action_mailer.raise_delivery_errors = false

# Enable locale fallbacks for I18n (makes lookups for any locale fall back to
# the I18n.default_locale when a translation cannot be found).
config.i18n.fallbacks = true

# Send deprecation notices to registered listeners.
config.active_support.deprecation = :notify

# Use default logging formatter so that PID and timestamp are not suppressed.
config.log_formatter = ::Logger::Formatter.new

# Use a different logger for distributed setups.
# require 'syslog/logger'
# config.logger = ActiveSupport::TaggedLogging.new(Syslog::Logger.new 'app-name')

if ENV["RAILS_LOG_TO_STDOUT"].present?
  logger      = ActiveSupport::Logger.new(STDOUT)
  logger.formatter = config.log_formatter
  config.logger  = ActiveSupport::TaggedLogging.new(logger)
end

# Do not dump schema after migrations.
config.active_record.dump_schema_after_migration = false
end

```

ADICIONANDO AS ALTERAÇÕES NO GIT

```

git status
git add .
git status
git commit -m "Usando bootstrap"
git push heroku master
heroku open

```

5 Receitas

Fettuccine



IMAGEM NÃO DISPONÍVEL

Ingredientes: 250g de Fettuccine, 70g bacon, 2...

Calorias: 280

Tempo de preparo: 00:13 min

Modo de preparo: Coloque o macarrão para cozinhar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinhar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.

Custo: \$50.00

Salada Light



Ingredientes: alface, agrião, brócolis, um mix de...

Calorias: Light (menos de 100 calorias)

Tempo de preparo: 00:10 min

Modo de preparo: Misture todas as folhas em uma tigela, tempere e sirva.

Custo: \$15.00

Salmão ao forno



Ingredientes: Sal marinho, Pimenta moída na hora,...

Calorias: 100

Tempo de preparo: 00:50 min

Modo de preparo: Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe temperando na geladeira por 1 h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis ou com cenourinha picada e cozida, refogada na manteiga

Custo: \$60.00

Filé de Frango a parmegiana



Ingredientes: 1/2 kg de filé de frango, 2 ovos...

Calorias: 160

Tempo de preparo: 00:35 min

Modo de preparo: Tempere os filetes de frango com o alho, suco de limão, ervas finas e sal a gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo. Coloque os filetes em um refratário. Cubra com a mussarela. Regue com o molho e leve ao forno para derreter a mussarela. Retire e sirva

Custo: \$40.00

Estrogofone de carne



Ingredientes: 1 kg de carne, 3 cebolas médias, sal e...

Calorias: 110

Tempo de preparo: 00:40 min

Modo de preparo: Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne. Acrescente o ketchup e o champignon. Desligue o fogo e acrescente o creme de leite sem soro. Coloque em uma forma refratária e decore com tempero e batata palha

Custo: \$50.00

Nova receita

AULA 18 – CUSTOMIZANDO A APLICAÇÃO

app/assets/javascripts.erb

```
// 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 bootstrap-sprockets
//= require activestorage
//= require turbolinks
//= require_tree .
```

app/views/layouts/application.erb

```
<!DOCTYPE html>
<html>
  <head>
    <title>Receitasnet</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' %>
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>

  <body>
    <%= render 'layouts/header' %>
    <%= yield %>
    <%= render 'layouts/footer' %>
  </body>
</html>
```

app/views/layouts/_header.erb

```
<nav class="navbar navbar-default">
  <div class="container-fluid">
    <!-- Brand and toggle get grouped for better mobile display -->
    <div class="navbar-header">
      <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
        <span class="sr-only">Toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <%= link_to image_tag('logo-receitas.png', size:'25', alt:'Receitasnet'), root_path, class:'navbar-brand' %>
      <%= link_to 'Receitasnet', root_path, class:'navbar-brand' %>
    </div>

    <!-- Collect the nav links, forms, and other content for toggling -->
    <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
      <ul class="nav navbar-nav navbar-right">
        <li><%= link_to 'Receitas', root_path %></li>
      </ul>
    </div><!-- /.navbar-collapse -->
  </div><!-- /.container-fluid -->
</nav>
```

app/views/layouts/_footer.erb

```
<footer class="footer"">
  <div class="container">
    <p class="text-muted">Receitas fáceis com Receitasnet.</p>
  </div>
</footer>
```

app/views/recipes/index.html.erb

```
<div class="container">

<div class="row">
  <div class="col-lg-12">
    <h1><%= pluralize(@recipes.size, "Receita", "Receitas") %>
      <small class="nova_receita"><%= link_to 'Nova receita', new_recipe_path %></small></h1>
    </div>
  </div>

<br>

<% @recipes.each do |recipe| %>
  <div class="row">
    <div class="col-md-5">
      <%= link_to imagem(recipe), recipe, class: 'imagem_direita' %>
    </div>
    <div class="col-md-7">
      <h2><%= link_to recipe.name, recipe %></h2>
      <p><strong>Ingredientes:</strong><%= truncate(recipe.stuff, length:40, separator:' ') %></p>
      <p><strong>Calorias:</strong><%= formata_caloria(recipe) %></p>
      <p><strong>Tempo de preparo:</strong><%= recipe.duration.to_formatted_s(:time) %> min</p>
      <p><strong>Custo:</strong><%= number_to_currency(recipe.cost) %></strong></p>
    </div>
  </div>
  <br>
<% end %>

</div>
```

app/assets/stylesheets/recipe.scss

```
.nova_receita{  
  float: right;  
}  
  
.imagem_direita{  
  float: right;  
  margin-right: 30px;  
}
```

ADICIONANDO AS ALTERAÇÕES NO GIT

```
git status  
git add .  
git status  
git commit -m "Melhorando o layout da aplicacao"  
git push heroku master  
heroku open
```

 Receitasnet Receitas

5 Receitas [Nova receita](#)


Fettuccine
Ingredientes: 250g de Fettuccine, 70g bacon, 2...
Calorias: 280
Tempo de preparo: 00:13 min
Custo: \$50.00









Salada Light
Ingredientes: alface, agrião, brócolis, um mix de...
Calorias: Light (menos de 100 calorias)
Tempo de preparo: 00:10 min
Custo: \$15.00

Salmão ao forno
Ingredientes: Sal marinho,Pimenta moida na hora,...
Calorias: 100
Tempo de preparo: 00:50 min
Custo: \$60.00

Filé de Frango a parmesana
Ingredientes: 1/2 kg de filé de frango, 2 ovos...
Calorias: 160
Tempo de preparo: 00:35 min
Custo: \$40.00

Estrogonofe de carne
Ingredientes: 1 kg de carne,3 cebolas médias, sal e...
Calorias: 110
Tempo de preparo: 00:40 min
Custo: \$50.00

AULA 19 – EMBELEZANDO OS FORMULÁRIOS

app/views/recipes/_form.html.erb

```
<%= form_for(@recipe) do |f| %>

  <p class="form-group">
    <%= f.label :Nome %><br>
    <%= f.text_field :name, class: 'form-control' %>
  </p>

  <p class="form-group">
    <%= f.label :Ingredientes %><br>
    <%= f.text_area :stuff, cols: 40, rows: 5, class: 'form-control' %>
  </p>

  <p class="form-group">
    <%= f.label :Tipo %><br>
    <%= f.select :kind, Recipe::KIND, prompt: 'Escolha uma opção', class: 'form-control' %>
  </p>

  <p class="form-group">
    <%= f.label :Calorias %><br>
    <%= f.number_field :calories, class: 'form-control' %>
  </p>

  <p class="form-group">
    <%= f.label :Modo_de_Prepago %><br>
    <%= f.text_area :prepare_mode, cols: 40, rows: 5, class: 'form-control' %>
  </p>

  <p class="form-group">
    <%= f.label :Tempo_de_Prepago %><br>
    <%= f.time_select :duration, class: 'form-control' %>
  </p>

  <p class="form-group">
    <%= f.label :Porções %><br>
    <%= f.text_field :portion, class: 'form-control' %>
  </p>

  <p class="form-group">
    <%= f.label :Custo %><br>
    <%= f.number_field :cost, class: 'form-control' %>
  </p>
```

```
<p class="form-group">
<%= f.label :Poster %><br>
<%= f.text_field :poster, placeholder: ".gif, .jpeg, .png", class: 'form-control' %>
</p>

<p>
<%= f.submit "Salvar receita", class:"btn btn-primary" %>
<%= link_to "Cancelar", root_path, class:"btn btn-default" %>
</p>
<% end %>
```

app/views/recipes/new.html.erb

```
<div class="container">
<div class="panel panel-default">
<div class="panel-heading"><h2>Criando receitas</h2></div>
<div class="panel-body">
<%= render 'form' %>
</div>
</div>
</div>
```

app/views/recipes/edit.html.erb

```
<div class="container">
<div class="panel panel-default">
<div class="panel-heading"><h2>Editando receitas</h2></div>
<div class="panel-body">
<%= render 'form' %>
</div>
</div>
</div>
```

app/views/recipes/show.html.erb

```
<div class="container">
<div class="jumbotron">
<article>
<head>
<h1><%= @recipe.name %></h1>
<%= imagem(@recipe) %>
</head>
<h2>Ingredientes</h2>
<p><%= @recipe.stuff %></p>
<h2>Calorias</h2>
<p><%= @recipe.calories %></p>
<h2>Modo de preparo</h2>
<p><%= @recipe.prepare_mode %></p>
<h2>Custo</h2>
<p><%= number_to_currency(@recipe.cost) %></p>
</article>

<%= link_to recipes_path, class:'btn btn-link' do %>
  <i class="glyphicon glyphicon-home"> Receitas</i>
<% end %>

<%= link_to edit_recipe_path(@recipe), class:'btn btn-link' do %>
  <i class="glyphicon glyphicon-edit"> Editar</i>
<% end %>

<%= link_to recipe_path(@recipe), method: :delete, data: { confirm: "Tem certeza que
deseja excluir esta receita?"}, class:'btn btn-link' do %>
  <i class="glyphicon glyphicon-remove-sign"> Deletar</i>
<% end %>

</div>
</div>
```

ADICIONANDO AS ALTERAÇÕES NO GIT

```
git status  
git add .  
git status  
git commit -m "Formatando forms e botoes"  
git push heroku master  
heroku open
```

The screenshot shows a web application for managing recipes. At the top, there's a header with a logo and the text 'Receitasnet'. On the right side of the header, there's a link labeled 'Receitas'. Below the header, the main content area has a title 'Editando receitas'. The form fields are organized into sections:

- Nome:** Estrogonofe de carne
- Ingredientes:** 1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino a gosto, 2 latas de creme de leite, 2 colheres (sopa) de ketchup
- Tipo:** Carnes
- Calorias:** 110
- Modo de preparo:** Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne. Acrescente o ketchup e o champignon. Desligue o fogo e acrescente o creme de leite sem soro. Coloque em uma forma refratária e decore com tempero e batata palha.
- Tempo de preparo:** 00 : 40
- Porções:** 5
- Custo:** 50,0
- Poster:** estrogonofe-de-carne.jpg

At the bottom of the form, there are two buttons: 'Salvar receita' (Save recipe) and 'Cancelar' (Cancel).

Receitas fáceis com Receitasnet.

Receitasnet Receitas

Criando receitas

Nome

Ingredientes

Tipo
Escolha uma opção ▾

Calorias

Modo de preparo

Tempo de preparo
00 ▾ : 26 ▾

Porções

Custo

Poster
.gif, .jpeg, .png

Salvar receita **Cancelar**

Receitas fáceis com Receitasnet.

Receitasnet Receitas

Estrogonofe de carne



Ingredientes

1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino a gosto, 2 latas de creme de leite, 2 colheres (sopa) de ketchup

Calorias

110

Modo de preparo

Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne. Acrescente o ketchup e o champignon. Desligue o fogo e acrescente o creme de leite sem soro. Coloque em uma forma refratária e decore com tempero e batata palha.

Custo

\$50.00

[Receitas](#) [Editar](#) [Deletar](#)

Receitas fáceis com Receitasnet.

AULA 20 – VALIDAÇÕES

rails c

```
receitasnet (master) $ rails c
Running via Spring preloader in process 813
Loading development environment (Rails 4.2.5.2)
2.2.1 :001 > r = Recipe.new
=> #<Recipe id: nil, name: nil, stuff: nil, calories: nil, created_at: nil, updated_at: nil, prepare_mode: nil, cost: nil, kind: nil, portion: nil, duration: nil, poster: nil>
2.2.1 :002 > r.valid?
=> false
2.2.1 :003 > r.errors
=> #<ActiveModel::Errors:0x007fc1d4bbb208 @base=#<Recipe id: nil, name: nil, stuff: nil, calories: nil, created_at: nil, updated_at: nil, prepare_mode: nil, cost: nil, kind: nil, portion: nil, duration: nil, poster: nil>, @messages={:name=>["can't be blank"], :stuff=>["can't be blank"], :duration=>["can't be blank"], :prepare_mode=>["can't be blank"], :calories=>["is not a number"], :poster=>["utilizar somente arquivos GIF,JPEG ou PNG."], :kind=>["is not included in the list"]}>
2.2.1 :004 > r.errors.full_messages
=> ["Name can't be blank", "Stuff can't be blank", "Duration can't be blank", "Prepare mode can't be blank", "Calories is not a number", "Poster utilizar somente arquivos GIF,JPEG ou PNG.", "Kind is not included in the list"]
2.2.1 :005 > r.errors.full_messages.to_sentence
=> "Name can't be blank, Stuff can't be blank, Duration can't be blank, Prepare mode can't be blank, Calories is not a number, Poster utilizar somente arquivos GIF,JPEG ou PNG., and Kind is not included in the list"
2.2.1 :006 > r.errors[:name]
=> ["can't be blank"]
```

app/controllers/recipes_controller.rb

```
class RecipesController < ApplicationController

  def index
    @recipes = Recipe.created_at
  end

  def show
    @recipe = Recipe.find(params[:id])
  end

  def new
    @recipe = Recipe.new
  end

  def create
    @recipe = Recipe.new(recipe_params)
    if @recipe.save
      redirect_to @recipe
    else
      render :new
    end
  end

  def edit
    @recipe = Recipe.find(params[:id])
  end
```

```
def update
  @recipe = Recipe.find(params[:id])
  if @recipe.update(recipe_params)
    redirect_to @recipe
  else
    render :edit
  end
end

def destroy
  @recipe = Recipe.find(params[:id])
  @recipe.destroy
  redirect_to recipes_url
end

private

def recipe_params
  params.require(:recipe).permit(:name, :stuff, :calories, :prepare_mode, :cost, :kind,
:portion, :duration, :poster)
end

end
```

app/assets/stylesheets/bootstrap_receita.css.scss

```
@import 'bootstrap-sprockets';
@import 'bootstrap';
```

- Reinicie o servidor.

app/models/recipe.rb

```
class Recipe < ApplicationRecord

  validates :name, :stuff, :duration, :prepare_mode, presence:true
  validates_length_of :stuff, maximum: 200
  validates_length_of :prepare_mode, maximum: 600
  validates_numericality_of :calories, greather_than_or_equal_to:0, presence:true
  validates_format_of :poster, allow_blank:true, with: /\w+\.(gif|jpeg|jpg|png)\z/i,
  message:" utilizar somente arquivos GIF,JPEG, JPG ou PNG."

  KIND = %w(Carnes Aves Peixes Massas Saladas Doces)
  validates_inclusion_of :kind , in: KIND

  def light?
    calories < 100
  end

  def self.created_at
    where('created_at <= ?', Time.now ).order('created_at desc')
  end

end
```

app/views/layouts/_errors.html.erb

```
<% if @recipe.errors.any? %>
<div class="alert alert-danger" role="alert">
<ul>
  <% @recipe.errors.full_messages.each do |message| %>
    <li><%= message %></li>
  <% end %>
</ul>
</div>
<% end %>
```

app/views/recipes/_form.html.erb

```
<%= form_for(@recipe) do |f| %>

<%= render 'layouts/errors', object: @recipe %>



<%= f.label :Nome %><br>
<%= f.text_field :name, class: 'form-control' %>



<%= f.label :Ingredientes %><br>
<%= f.text_area :stuff, cols: 40, rows: 5, class: 'form-control' %>



<%= f.label :Tipo %><br>
<%= f.select :kind, Recipe::KIND, prompt: 'Escolha uma opção', class: 'form-control' %>



<%= f.label :Calorias %><br>
<%= f.number_field :calories, class: 'form-control' %>



<%= f.label :Modo_de_Preparo %><br>
<%= f.text_area :prepare_mode, cols: 40, rows: 5, class: 'form-control' %>



<%= f.label :Tempo_de_Preparo %><br>
<%= f.time_select :duration, class: 'form-control' %>



<%= f.label :Porções %><br>
<%= f.text_field :portion, class: 'form-control' %>



<%= f.label :Custo %><br>
<%= f.number_field :cost, class: 'form-control' %>



<%= f.label :Poster %><br>
<%= f.text_field :poster, placeholder: ".gif, .jpeg, .png", class: 'form-control' %>


```

```

<p>
  <%= f.submit "Salvar receita", class:"btn btn-primary" %>
  <%= link_to "Cancelar", root_path, class:"btn btn-default" %>
</p>
<% end %>

```

ADICIONANDO AS ALTERAÇÕES NO GIT

```

git status
git add .
git status
git commit -m "Criando validadores nos modelos"
git push heroku master
heroku open

```

Criando receitas

- Name can't be blank
- Stuff can't be blank
- Prepare mode can't be blank
- Calories is not a number
- Kind is not included in the list

Nome

Ingredientes

Tipo

Calorias

Modo de preparo

Tempo de preparo

Porções

Custo

Poster

Salvar receita **Cancelar**

Editando receitas

- Stuff can't be blank

Nome**Ingredientes****Tipo****Calorias****Modo de preparo****Tempo de preparo****Porções****Custo****Poster**

AULA 21 – FALANDO OUTRAS LÍNGUAS

Para internacionalização, o Ruby utiliza a gem i18n.

Acesse a URL:

<https://github.com/svenfuchs/rails-i18n/blob/master/rails/locale/pt-BR.yml>

config/locales/pt-BR.yml

```
pt-BR:
  activerecord:
    models:
      recipe: Receita
    attributes:
      recipe:
        name: Nome da Receita
        stuff: Ingredientes
        calories: Calorias
        kind: Tipo
        prepare_mode: Modo de preparo
        cost: Custo
        portion: Porção
        duration: Tempo de preparo
        poster: Poster
    errors:
      messages:
        record_invalid: 'A validação falhou: %{errors}'
        restrict_dependent_destroy:
          has_one: Não é possível excluir o registro pois existe um %{record} dependente
          has_many: Não é possível excluir o registro pois existem %{record} dependentes
    date:
      abbr_day_names:
        - dom
        - seg
        - ter
        - qua
        - qui
        - sex
        - sáb
      abbr_month_names:
        -
        - jan
        - fev
        - mar
        - abr
        - mai
        - jun
```

```
- jul
- ago
- set
- out
- nov
- dez
day_names:
- domingo
- segunda-feira
- terça-feira
- quarta-feira
- quinta-feira
- sexta-feira
- sábado
formats:
default: "%d/%m/%Y"
long: "%d de %B de %Y"
short: "%d de %B"
month_names:
-
- janeiro
- fevereiro
- março
- abril
- maio
- junho
- julho
- agosto
- setembro
- outubro
- novembro
- dezembro
order:
- :day
- :month
- :year
datetime:
distance_in_words:
about_x_hours:
one: aproximadamente 1 hora
other: aproximadamente %{count} horas
about_x_months:
one: aproximadamente 1 mês
other: aproximadamente %{count} meses
about_x_years:
one: aproximadamente 1 ano
other: aproximadamente %{count} anos
almost_x_years:
one: quase 1 ano
other: quase %{count} anos
half_a_minute: meio minuto
less_than_x_seconds:
```

```
one: menos de 1 segundo
other: menos de %{count} segundos
less_than_x_minutes:
  one: menos de um minuto
  other: menos de %{count} minutos
over_x_years:
  one: mais de 1 ano
  other: mais de %{count} anos
x_seconds:
  one: 1 segundo
  other: "%{count} segundos"
x_minutes:
  one: 1 minuto
  other: "%{count} minutos"
x_days:
  one: 1 dia
  other: "%{count} dias"
x_months:
  one: 1 mês
  other: "%{count} meses"
x_years:
  one: 1 ano
  other: "%{count} anos"
prompts:
  second: Segundo
  minute: Minuto
  hour: Hora
  day: Dia
  month: Mês
  year: Ano
errors:
  format: "%{attribute} %{message}"
  messages:
    accepted: deve ser aceito
    blank: não pode ficar em branco
    confirmation: não é igual a %{attribute}
    empty: não pode ficar vazio
    equal_to: deve ser igual a %{count}
    even: deve ser par
    exclusion: não está disponível
    greater_than: deve ser maior que %{count}
    greater_than_or_equal_to: deve ser maior ou igual a %{count}
    inclusion: não está incluído na lista
    invalid: não é válido
    less_than: deve ser menor que %{count}
    less_than_or_equal_to: deve ser menor ou igual a %{count}
    model_invalid: 'A validação falhou: %{errors}'
    not_a_number: não é um número
    not_an_integer: não é um número inteiro
    odd: deve ser ímpar
    other_than: deve ser diferente de %{count}
    present: deve ficar em branco
```

```
required: é obrigatório(a)
taken: já está em uso
too_long:
  one: 'é muito longo (máximo: 1 caracter)'
  other: 'é muito longo (máximo: %{count} caracteres)'
too_short:
  one: 'é muito curto (mínimo: 1 caracter)'
  other: 'é muito curto (mínimo: %{count} caracteres)'
wrong_length:
  one: não possui o tamanho esperado (1 caracter)
  other: não possui o tamanho esperado (%{count} caracteres)
template:
  body: 'Por favor, verifique o(s) seguinte(s) campo(s):'
  header:
    one: 'Não foi possível gravar %{model}: 1 erro'
    other: 'Não foi possível gravar %{model}: %{count} erros'
helpers:
  select:
    prompt: Por favor selecione
  submit:
    create: Criar %{model}
    submit: Salvar %{model}
    update: Atualizar %{model}
number:
  currency:
    format:
      delimiter: "."
      format: "%u %n"
      precision: 2
      separator: ","
      significant: false
      strip_insignificant_zeros: false
      unit: R$
    format:
      delimiter: "."
      precision: 3
      separator: ","
      significant: false
      strip_insignificant_zeros: false
  human:
    decimal_units:
      format: "%n %u"
    units:
      billion:
        one: bilhão
        other: bilhões
      million:
        one: milhão
        other: milhões
      quadrillion:
        one: quatrilhão
        other: quatrilhões
```

```
thousand: mil
trillion:
  one: trilhão
  other: trilhões
  unit: ""
format:
  delimiter: ""
  precision: 3
  significant: true
  strip_insignificant_zeros: true
storage_units:
  format: "%n %u"
  units:
    byte:
      one: Byte
      other: Bytes
    eb: EB
    gb: GB
    kb: KB
    mb: MB
    pb: PB
    tb: TB
percentage:
  format:
    delimiter: "."
    format: "%n%"
precision:
  format:
    delimiter: "."
support:
  array:
    last_word_connector: " e "
    two_words_connector: " e "
    words_connector: ","
time:
  am: ""
  formats:
    default: "%a, %d de %B de %Y, %H:%M:%S %z"
    long: "%d de %B de %Y, %H:%M"
    short: "%d de %B, %H:%M"
  pm: ""
```

config/application.rb

```
require_relative 'boot'

require 'rails/all'

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

module Receitalistanet
  class Application < Rails::Application
    # Initialize configuration defaults for originally generated Rails version.
    config.load_defaults 5.2
    config.time_zone = 'Brasilia'

    config.i18n.default_locale = "pt-BR"
    I18n.config.available_locales = "pt-BR"

    # 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.
  end
end
```

- Reinicie o servidor.

app/views/recipes/_form.html.erb

```
<%= form_for(@recipe) do |f| %>

<%= render 'layouts/errors', object: @recipe %>



<%= f.label :Nome %><br>
<%= f.text_field :name, class: 'form-control' %>



<%= f.label :Ingredientes %><br>
<%= f.text_area :stuff, cols: 40, rows: 5, class: 'form-control' %>



<%= f.label :Tipo %><br>
<%= f.select :kind, Recipe::KIND, prompt: 'Escolha uma opção', class: 'form-control' %>



<%= f.label :Calorias %><br>
<%= f.number_field :calories, class: 'form-control' %>



<%= f.label :Modo_de_Preparo %><br>
<%= f.text_area :prepare_mode, cols: 40, rows: 5, class: 'form-control' %>



<%= f.label :Tempo_de_Preparo %><br>
<%= f.time_select :duration, class: 'form-control' %>



<%= f.label :Porções %><br>
<%= f.text_field :portion, class: 'form-control' %>



<%= f.label :Custo %><br>
<%= f.number_field :cost, class: 'form-control' %>



<%= f.label :Poster %><br>
<%= f.text_field :poster, placeholder: ".gif, .jpeg, .png", class: 'form-control' %>


```

```
<p>
  <%= f.submit class:"btn btn-primary" %>
  <%= link_to "Cancelar", root_path, class:"btn btn-default" %>
</p>
<% end %>
```

ADICIONANDO AS ALTERAÇÕES NO GIT

```
git status
git add .
git status
git commit -m "Falando outras linguas"
git push heroku master
heroku open
```

AULA 22 – FLASH MESSAGES

`app/controllers/recipes_controller.rb`

```
class RecipesController < ApplicationController

  def index
    @recipes = Recipe.created_at
  end

  def show
    @recipe = Recipe.find(params[:id])
  end

  def new
    @recipe = Recipe.new
  end

  def create
    @recipe = Recipe.new(recipe_params)
    if @recipe.save
      redirect_to @recipe, notice: "Receita criada com sucesso!"
    else
      render :new
    end
  end

  def edit
    @recipe = Recipe.find(params[:id])
  end

  def update
    @recipe = Recipe.find(params[:id])
    if @recipe.update(recipe_params)
      redirect_to @recipe, notice: "Receita atualizada com sucesso!"
    else
      render :edit
    end
  end

  def destroy
    @recipe = Recipe.find(params[:id])
    @recipe.destroy
    redirect_to recipes_url, alert: "Receita apagada com sucesso!"
  end
end
```

```
private

def recipe_params
  params.require(:recipe).permit(:name, :stuff, :calories, :prepare_mode, :cost, :kind,
:portion, :duration, :poster)
end

end
```

app/views/layouts/applications.erb

```
<!DOCTYPE html>
<html>
<head>
  <title>Receitasnet</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' %>
  <meta name="viewport" content="width=device-width, initial-scale=1">
</head>

<body>

  <%= render 'layouts/header' %>

  <%= render 'layouts/flash' %>

  <%= yield %>

  <%= render 'layouts/footer' %>

</body>
</html>
```

app/views/layouts/_flash.erb

```
<% if flash[:notice] %>
<div class='container'>
  <div class="alert alert-success alert-dismissible" role="alert">
    <button type="button" class="close" data-dismiss="alert" aria-label="Close"><span aria-hidden="true">&times;</span></button>
    <%= flash[:notice] %>
  </div>
</div>
<% elsif flash[:alert] %>
<div class='container'>
  <div class="alert alert-danger alert-dismissible" role="alert">
    <button type="button" class="close" data-dismiss="alert" aria-label="Close"><span aria-hidden="true">&times;</span></button>
    <%= flash[:alert] %>
  </div>
</div>
<% 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 Puma as the app server
gem 'puma', '~> 3.11'
# Bootstrap
gem 'bootstrap-sass'
# 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 'ductape'
# 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
```

```

# Use jquery as the JavaScript library
gem 'jquery-rails'
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 ActiveStorage 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
  # Use mysql as the database for Active Record
  gem 'mysql2', '>= 0.4.4', '< 0.6.0'
  # Access an interactive console on exception pages or by calling 'console' anywhere in the
  # code.
  gem 'web-console', '>= 3.3.0'
  # Call 'byebug' anywhere in the code to stop execution and get a debugger console
  gem 'byebug', platforms: [:mri, :mingw, :x64_mingw]
end

group :production do
  gem "pg"
  gem 'rails_12factor'
end

group :test do
  # Adds support for Capybara system testing and selenium driver
  gem 'capybara', '>= 2.15'
  gem 'selenium-webdriver'
  # Easy installation and use of chromedriver to run system tests with Chrome
  gem 'chromedriver-helper'
end

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

```

bundle install –without production

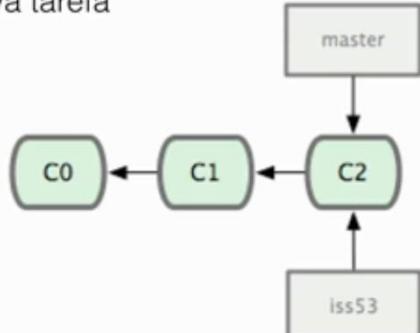
app/assets/javascripts/application.js

```
// 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 jquery_ujs
//= require rails-ujs
//= require bootstrap-sprockets
//= require activestorage
//= require turbolinks
//= require_tree .
```

Criando um branch para as flash messages

GIT

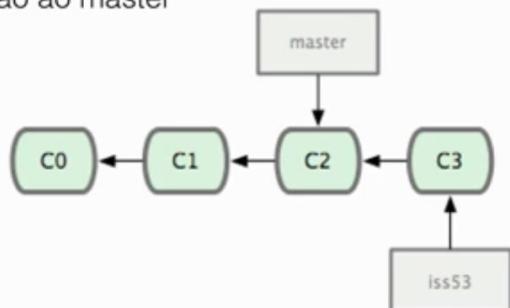
Criar um novo branch para
uma nova tarefa



git checkout -b <novo_branch>

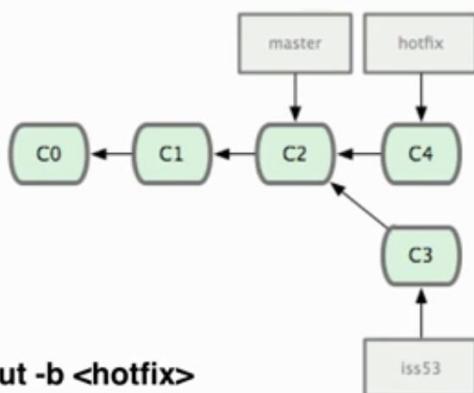
GIT

Novo branch está avançado em
relação ao master

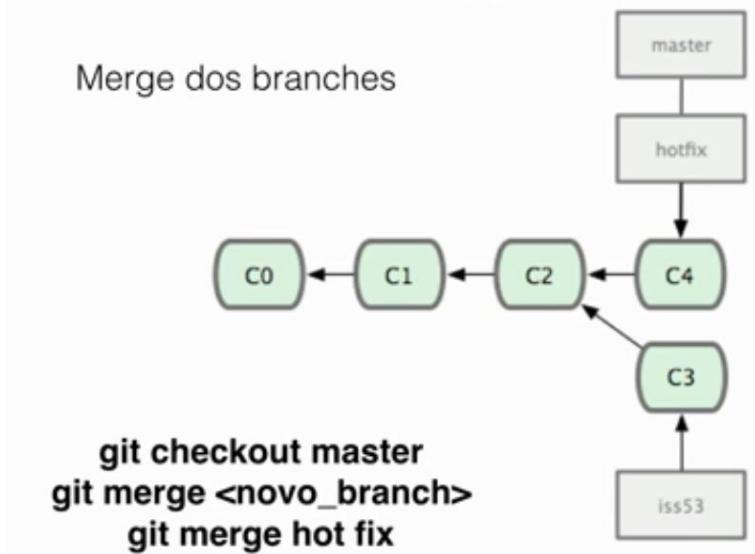


GIT

Novo branch hotfix criado



GIT



git checkout –b flash

```
C:\Sites\receitasnet>git checkout -b flash
Switched to a new branch 'flash'
M      Gemfile
M      Gemfile.lock
M      app/assets/javascripts/application.js
M      app/controllers/recipes_controller.rb
M      app/views/layouts/application.html.erb
```

ADICIONANDO AS ALTERAÇÕES NO GIT

git status

```
C:\Sites\receitasnet>git status
On branch flash
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   Gemfile
    modified:   Gemfile.lock
    modified:   app/assets/javascripts/application.js
    modified:   app/controllers/recipes_controller.rb
    modified:   app/views/layouts/application.html.erb

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    app/views/layouts/_flash.html.erb

no changes added to commit (use "git add" and/or "git commit -a")
```

```
git add .
```

```
C:\Sites\receitasnet>git add .
warning: LF will be replaced by CRLF in Gemfile.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in Gemfile.lock.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in app/assets/javascripts/application.js.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in app/controllers/recipes_controller.rb.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in app/views/layouts/application.html.erb.
The file will have its original line endings in your working directory
```

```
git status
```

```
C:\Sites\receitasnet>git status
On branch flash
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   Gemfile
    modified:   Gemfile.lock
    modified:   app/assets/javascripts/application.js
    modified:   app/controllers/recipes_controller.rb
    new file:   app/views/layouts/_flash.html.erb
    modified:   app/views/layouts/application.html.erb
```

```
git commit -m "Falando outras linguas"
```

```
C:\Sites\receitasnet>git commit -m "Criando flash messages"
[flash 4b1e519] Criando flash messages
 6 files changed, 30 insertions(+), 5 deletions(-)
 create mode 100644 app/views/layouts/_flash.html.erb
```

```
git checkout master
```

```
C:\Sites\receitasnet>git checkout master
Switched to branch 'master'
```

```
git merge flash
```

```
C:\Sites\receitasnet>git merge flash
Updating 33c1efb..4b1e519
Fast-forward
  Gemfile                  |  3 +-+
  Gemfile.lock              |  5 +++++
  app/assets/javascripts/application.js |  4 +-+-
  app/controllers/recipes_controller.rb |  6 +--+-
  app/views/layouts/_flash.html.erb    | 15 ++++++-----+
  app/views/layouts/application.html.erb|  2 ++
  6 files changed, 30 insertions(+), 5 deletions(-)
  create mode 100644 app/views/layouts/_flash.html.erb
```

```
git branch -d flash
```

```
C:\Sites\receitasnet>git branch -d flash
Deleted branch flash (was 4b1e519).
```

```
git push heroku master
```

```
heroku open
```

Receitasnet Receitas

Receita atualizada com sucesso! ×

Estrogonofe de carne



Ingredientes

1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino a gosto, 2 latas de creme de leite, 2 colheres (sopa) de ketchup

Calorias

110

Modo de preparo

Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne. Acrescente o ketchup e o champignon. Desligue o fogo e acrescente o creme de leite sem soro. Coloque em uma forma refratária e decore com tempero e batata palha.

Custo

R\$ 50,00

[Receitas](#) [Editar](#) [Deletar](#)

Receitas fáceis com Receitasnet.

AULA 23 – RELACIONAMENTO ENTRE TABELAS

```
rails g resource comment name:string rating:integer comment:text recipe:references
```

```
C:\Sites\receitasnet>rails g resource comment name:string rating:integer comment:text recipe:references
  invoke  active_record
  create    db/migrate/20191013201138_create_comments.rb
  create    app/models/comment.rb
  invoke  test_unit
  create    test/models/comment_test.rb
  create    test/fixtures/comments.yml
  invoke  controller
  create    app/controllers/comments_controller.rb
  invoke  erb
  create    app/views/comments
  invoke  test_unit
  create    test/controllers/comments_controller_test.rb
  invoke  helper
  create    app/helpers/comments_helper.rb
  invoke  test_unit
  invoke  assets
  invoke  coffee
  create    app/assets/javascripts/comments.coffee
  invoke  scss
  create    app/assets/stylesheets/comments.scss
  invoke  resource_route
  route    resources :comments
```

db/migrate/20191013201138_create_comments.rb

```
class CreateComments < ActiveRecord::Migration[5.2]
  def change
    create_table :comments do |t|
      t.string :name
      t.integer :rating
      t.text :comment
      t.references :recipe, foreign_key: true

      t.timestamps
    end
  end
end
```

rake db:migrate

```
C:\Sites\receitasnet>rake db:migrate
== 20191013201138 CreateComments: migrating =====
-- create_table(:comments)
  -> 0.5759s
== 20191013201138 CreateComments: migrated (0.5767s) =====
```

Tabela comments

Colunas:	Adicionar	Remover	Mover para cima	Mover para baixo			
#	Nome	Tipo de dados	Tamanho/Ite...	Unsign...	Permitir...	Zerofill	Padrão
1	id	BIGINT	20	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	name	VARCHAR	255	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	NULL
3	rating	INT	11	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
4	comment	TEXT		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
5	recipe_id	BIGINT	20	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
6	created_at	DATETIME		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Nenhum padrão
7	updated_at	DATETIME		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Nenhum padrão

Modelos



app/models/comments.rb

```
class Comment < ApplicationRecord
  belongs_to :recipe
end
```

app/models/recipe.rb

```
class Recipe < ApplicationRecord

  validates :name, :stuff, :duration, :prepare_mode, presence:true
  validates_length_of :stuff, maximum: 200
  validates_length_of :prepare_mode, maximum: 600
  validates_numericality_of :calories, greather_than_or_equal_to:0, presence:true
  validates_format_of :poster, allow_blank:true, with: /\w+\.(gif|jpeg|jpg|png)\z/i,
  message:" utilizar somente arquivos GIF,JPEG, JPG ou PNG."

  KIND = %w(Carnes Aves Peixes Massas Saladas Doces)
  validates_inclusion_of :kind , in: KIND

  has_many :comments , dependent: :destroy

  def light?
    calories < 100
  end

  def self.created_at
    where('created_at <= ?', Time.now ).order('created_at desc')
  end

end
```

rails c

Recipe

receita = Recipe.first

```
C:\Sites\receitasnet>rails c
Loading development environment (Rails 5.2.3)
irb(main):001:0> Recipe
=> Recipe (call 'Recipe.connection' to establish a connection)
irb(main):002:0> receita = Recipe.first
  (0.5ms)  SET NAMES utf8, @@SESSION.sql_mode = CONCAT(CONCAT(@@sql_mode, 'STRICT_ALL_TABLES'), ',NO_AUTO_VALUE_ON_ZERO'), @@SESSION.sql_auto_is_null = 0, @@SESSION.wait_timeout = 2147483
  (0.5ms)  SET NAMES utf8, @@SESSION.sql_mode = CONCAT(CONCAT(@@sql_mode, 'STRICT_ALL_TABLES'), ',NO_AUTO_VALUE_ON_ZERO'), @@SESSION.sql_auto_is_null = 0, @@SESSION.wait_timeout = 2147483
  Recipe Load (0.3ms)  SELECT `recipes`.* FROM `recipes` ORDER BY `recipes`.`id` ASC LIMIT 1
=> #<Recipe id: 1, name: "Estrogonofe de carne", stuff: "1 kg de carne,3 cebolas médias, sal e pimenta-do-rebisco...", calories: 110, created_at: "2019-10-12 14:51:06", updated_at: "2019-10-13 17:46:16", prepare_mode: "Em uma panela, adicione o óleo, a carne, a cebola,...", cost: 50, kind: "Carnes", portion: "5", duration: "2016-05-05 03:40:00", poster: "estrogonofe-de-carne.jpg">
```

Retornando os comentários desta receita

receita.comments

```
irb(main):003:0> receita.comments
Comment Load (95.9ms)  SELECT `comments`.* FROM `comments` WHERE `comments`.`recipe_id` = 1 LIMIT 11
=> #<ActiveRecord::Associations::CollectionProxy []>
```

Criando um comentário

comentario = Comment.new(name: "Jose Silva", rating: 8, comment: "Boa receita!")

```
irb(main):004:0> comentario = Comment.new(name: "Jose Silva", rating: 8, comment: "Boa receita!")
=> #<Comment id: nil, name: "Jose Silva", rating: 8, comment: "Boa receita!", recipe_id: nil, created_at: nil, updated_at: nil>
```

comentario.recipe_id

```
irb(main):005:0> comentario.recipe_id
=> nil
```

Retorna vazio porque ainda não foi feita nenhuma associação para essa consulta.

Fazendo a associação entre o comentário e a receita

comentario.recipe_id = receita.id

ou

comentario.recipe = receita

```
irb(main):006:0> comentario.recipe_id = receita.id
=> 1
```

Receitas que possuem comentários

comentario.recipe

```
irb(main):007:0> comentario.recipe
Recipe Load (1.7ms)  SELECT `recipes`.* FROM `recipes` WHERE `recipes`.`id` = 1 LIMIT 1
=> #<Recipe id: 1, name: "Estrogonofe de carne", stuff: "1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino", calories: 110, created_at: "2019-10-12 14:51:06", updated_at: "2019-10-13 17:46:16", prepare_mode: "Em uma panela, adicione o óleo, a carne, a cebola, ...", cost: 50, kind: "Carnes", portion: "5", duration: "2016-05-05 03:40:00", poster: "estrogonofe-de-carne.jpg">
```

Salvando o comentário

comentario.save

```
irb(main):008:0> comentario.save
  (0.7ms)  BEGIN
Comment Create (1.8ms)  INSERT INTO `comments` (`name`, `rating`, `comment`, `recipe_id`, `created_at`, `updated_at`) VALUES ('Jose Silva', 8, 'Boa receita!', 1, '2019-10-13 22:21:44', '2019-10-13 22:21:44')
  (159.7ms)  COMMIT
=> true
```

Tabela comments

receitasnet_development.comments: 1 registros totais (aproximadamente)							Próx
id	name	rating	comment	recipe_id	created_at	updated_at	
1	Jose Silva	8	Boa receita!	1	2019-10-13 22:21:44	2019-10-13 22:21:44	

Listando os comentários da primeira receita

receita.reload
receita.comments

```
irb(main):009:0> receita.comments
Comment Load (1.1ms)  SELECT `comments`.* FROM `comments` WHERE `comments`.`recipe_id` = 1 LIMIT 11
=> #<ActiveRecord::Associations::CollectionProxy:0x007fbc40101000 [#<Comment id: 1, name: "Jose Silva", rating: 8, comment: "Boa receita!", recipe_id: 1, created_at: "2019-10-13 22:21:44", updated_at: "2019-10-13 22:21:44">>]
```

Criando um novo registro a partir da tabela receitas

comentario=receita.comments.new(name: "Maria Jose", rating: 9, comment: "Muito bom!")

```
irb(main):010:0> comentario=receita.comments.new(name: "Maria Jose", rating: 9, comment: "Muito bom!")
=> #<Comment id: nil, name: "Maria Jose", rating: 9, comment: "Muito bom!", recipe_id: 1, created_at: nil, updated_at: nil>
```

comentario.save

```
irb(main):011:0> comentario.save
  (0.7ms)  BEGIN
Comment Create (0.9ms)  INSERT INTO `comments` (`name`, `rating`, `comment`, `recipe_id`, `created_at`, `updated_at`) VALUES ('Maria Jose', 9, 'Muito bom!', 1, '2019-10-13 22:37:35', '2019-10-13 22:37:35')
  (164.4ms)  COMMIT
=> true
```

receitasnet_development.comments: 2 registros totais (aproximadamente)							Próximo
id	name	rating	comment	recipe_id	created_at	updated_at	
1	Jose Silva	8	Boa receita!	1	2019-10-13 22:21:44	2019-10-13 22:21:44	
2	Maria Jose	9	Muito bom!	1	2019-10-13 22:37:35	2019-10-13 22:37:35	

VALIDAÇÕES NO MODELO COMMENT

app/models/comment.rb

```
class Comment < ApplicationRecord
  validates :name, :rating, :comment, presence: true
  validates_length_of :comment, maximum: 200

  RATING = [0,1,2,3,4,5]
  validates_inclusion_of :rating, in: RATING

  belongs_to :recipe
end
```

db/seeds.rb

```
Recipe.create!([
  {
    name: "Estrogonofe de carne",
    stuff: "1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino a gosto, 2 latas de creme de leite, 2 colheres (sopa) de ketchup",
    calories: 110,
    kind: "Carnes",
    prepare_mode: "Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne. Acrescente o ketchup e o champignon. Desligue o fogo e acrescente o creme de leite sem soro. Coloque em uma forma refratária e decore com tempero e batata palha. ",
    portion: "5",
    duration: "2016-05-06 00:40:00",
    cost: 50.00,
    poster: "estrogonofe-de-carne.jpg"
  },
  {
    name: "Filé de Frango a parmegiana",
    stuff: "1/2 kg de filé de frango, 2 ovos batidos com 1 pitada de sal, 2 xícaras (chá) de farinha de rosca, 1 caixinha de molho de tomate pronto, mussarela",
    calories: 160,
    kind: "Aves",
    prepare_mode: "Tempere os filés de frango com o alho, suco de limão, ervas finas e sal a gosto. Passe nos ovos batidos e na farinha de rosca. Frite em óleo. Coloque os filés em um refratório. Cubra com a mussarela. Regue com o molho e leve ao forno para derreter a mussarela. Retire e sirva",
    portion: "8",
    duration: "2016-05-06 0:35:00",
    cost: 40.00,
    poster: "file-frango-parmegiana.jpeg"
  },
  {
    name: "Salmão ao forno",
    stuff: "Sal marinho, Pimenta moída na hora, Azeite extra virgem, 2 filés de salmão 1.1kg com pele, sem escamas e espinhas, 2 limões",
    calories: 100,
    kind: "Peixes",
    prepare_mode: "Retire as espinhas do filé, passe o sal grosso e as ervas. Regue com o azeite e deixe temperando na geladeira por 1 h. Leve ao forno sem cobrir, de 180° a 200°, por no máximo 30 minutos, para não ressecar. Sirva com arroz de brócolis ou com cenourinha picada e cozida, refogada na manteiga",
    portion: "4",
    duration: "2016-05-06 0:50:00",
    cost: 60.00,
    poster: "salmao-forno.jpeg"
  },
  {
    name: "Salada Light",
    stuff: "alface, agrião, brócolis, um mix de folhas",
  }
])
```

```

        calories: 90,
        kind: "Saladas",
        prepare_mode: "Misture todas as folhas em uma tigela, tempere e sirva.",
        portion: "4",
        duration: "2016-05-06 0:10:00",
        cost:15.00,
        poster: "salada-light.jpeg"
    },
    {
        name: "Fettuccine",
        stuff: "250g de Fettuccine, 70g bacon, 2 caixas de creme de leite, 3 colheres (sopa) de queijo ralado de saquinho (parmesão), 2 dentes de alho, 1 colher (de sopa) de manteiga/margarina, sal a gosto.",
        calories: 280,
        kind: "Massas",
        prepare_mode: "Coloque o macarrão para cozinar em água salgada, mas não deixe que fique muito mole (pois vai terminar de cozinar no molho). Enquanto o macarrão cozinha frite o bacon (ou o que desejar) e reserve, na mesma panela adicione manteiga e refogue rapidamente o alho triturado, junte o creme de leite e misture em fogo baixo até que comece a querer levantar fervura. Adicione o queijo ralado, misture rapidamente para derreter um pouco o queijo. Adicione o fettuccine e o bacon e misture bem.",
        portion: "4",
        duration: "2016-05-06 0:13:00",
        cost:50.00,
        poster: ""
    },
])
receita = Recipe.find_by(name: 'Estrogonofe de carne')
receita.comments.create!(name: "Paulo Xavier", rating: 5, comment: "Delicioso")
receita.comments.create!(name: "Joana Santana", rating: 4, comment: "Muito bom.")
receita.comments.create!(name: "Augusto Barros", rating: 4, comment: "Gostoso")
receita = Recipe.find_by(name: 'Salmão ao forno')
receita.comments.create!(name: "Marta Freitas", rating: 5, comment: "O melhor que já comi.")

```

```
rake db:reset
```

```
C:\Sites\receitasnet>rake db:reset
Dropped database 'receitasnet_development'
Dropped database 'receitasnet_test'
Created database 'receitasnet_development'
Created database 'receitasnet_test'
-- create_table("comments", {:options=>"ENGINE=InnoDB DEFAULT CHARSET=utf8", :force=>:cascade})
  -> 0.2863s
-- create_table("recipes", {:options=>"ENGINE=InnoDB DEFAULT CHARSET=utf8", :force=>:cascade})
  -> 0.2623s
-- add_foreign_key("comments", "recipes")
  -> 0.6680s
-- create_table("comments", {:options=>"ENGINE=InnoDB DEFAULT CHARSET=utf8", :force=>:cascade})
  -> 0.2335s
-- create_table("recipes", {:options=>"ENGINE=InnoDB DEFAULT CHARSET=utf8", :force=>:cascade})
  -> 0.2402s
-- add_foreign_key("comments", "recipes")
  -> 0.6779s
```

Tabela comments

receitasnet_development.comments: 4 registros totais (aproximadamente)							Próximo	MM
✍ id	name	rating	comment	👉 recipe_id	created_at	updated_at		
1	Paulo Xavier	5	Delicioso	1	2019-10-13 22:56:50	2019-10-13 22:56:50		
2	Joana Santana	4	Muito bom.	1	2019-10-13 22:56:50	2019-10-13 22:56:50		
3	Augusto Barros	4	Gostoso	1	2019-10-13 22:56:50	2019-10-13 22:56:50		
4	Marta Freitas	5	O melhor que já comi.	3	2019-10-13 22:56:50	2019-10-13 22:56:50		

ADICIONANDO AS ALTERAÇÕES NO GIT

```
git status
git add .
git status
git commit -m "Criando novo modelo e seus relacionamentos"
git push heroku master
```

Rodando as migrações no heroku

```
heroku run rake db:migrate
```

Abrindo a aplicação

```
heroku open
```

AULA 24 – RELACIONANDO RECURSOS

1. Criar um relacionamento recipe → comment
2. action index no CommentController
3. template index.html.erb
4. links para os comentários da página show

CONFIGURANDO O ARQUIVO DE ROTAS

config/routes.rb

```
Rails.application.routes.draw do  
  
  root 'recipes#index'  
  
  resources :recipes do  
    resources :comments  
  end  
  
end
```

rake routes

```
C:\Sites\receitasnet>rake routes  
      Prefix Verb   URI Pattern                               Controller#Action  
        root GET    /                                         recipes#index  
recipe_comments GET   /recipes/:recipe_id/comments(.:format) comments#index  
                  POST   /recipes/:recipe_id/comments(.:format) comments#create  
new_recipe_comment GET   /recipes/:recipe_id/comments/new(.:format) comments#new  
edit_recipe_comment GET   /recipes/:recipe_id/comments/:id/edit(.:format) comments#edit  
recipe_comment GET   /recipes/:recipe_id/comments/:id(.:format) comments#show  
                 PATCH  /recipes/:recipe_id/comments/:id(.:format) comments#update  
                 PUT    /recipes/:recipe_id/comments/:id(.:format) comments#update  
                 DELETE /recipes/:recipe_id/comments/:id(.:format) comments#destroy  
recipes GET   /recipes(.:format)                           recipes#index  
       POST  /recipes(.:format)                           recipes#create  
new_recipe GET   /recipes/new(.:format)                      recipes#new  
edit_recipe GET   /recipes/:id/edit(.:format)                   recipes#edit  
recipe GET   /recipes/:id(.:format)                          recipes#show  
          PATCH  /recipes/:id(.:format)                      recipes#update  
          PUT    /recipes/:id(.:format)                      recipes#update  
          DELETE /recipes/:id(.:format)                      recipes#destroy  
rails_service_blob GET   /rails/active_storage/blobs/:signed_id/*filename(.:format) active_storage/blobs#show  
rails_blob_representation GET  /rails/active_storage/representations/:signed_blob_id/:variation_key/*filename(.:format) active_storage/representations#show  
rails_disk_service GET   /rails/active_storage/disk/:encoded_key/*filename(.:format) active_storage/disk#show  
update_rails_disk_service PUT  /rails/active_storage/disk/:encoded_token(.:format) active_storage/disk#update  
rails_direct_uploads POST /rails/active_storage/direct_uploads(.:format) active_storage/direct_uploads#create
```

app/controllers/comments_controller.rb

```
class CommentsController < ApplicationController
  def index
    @recipe = Recipe.find(params[:recipe_id])
    @comments = @recipe.comments
  end
end
```

app/views/comments/index.html.erb

```
<div class="container">

<% if @recipe.comments.size > 0 %>
  <div class="jumbotron">
    <ul>
      <% @comments.each do |comment| %>
        <li>
          <p>
            <%= comment.name %> <small class="small_font_rating"><%= pluralize(comment.rating, 'estrela', 'estrelas') %></small>
            <small class="small_font"> - <%= time_ago_in_words(comment.created_at) %>
            atrás</small>
          </p>
          <p>
            <%= comment.comment %>
          </p>
        </li>
      <% end %>
    </ul>
  </div>
<% else %>
  <div class="jumbotron">
    <h3 class="text-center">Não existem comentários!</h3>
  </div>
<% end %>

<%= link_to recipes_path, class:'btn btn-link' do %>
  <i class="glyphicon glyphicon-home"> Receitas</i>
<% end %>

</div>

<br>
```

app/views/comments/show.html.erb

```
<div class="container">
<div class="jumbotron">
<article>
<head>
  <h1><%= @recipe.name %></h1>
  <%= imagem(@recipe) %>
</head>
<h2>Ingredientes</h2>
<p><%= @recipe.stuff %></p>
<h2>Calorias</h2>
<p><%= @recipe.calories %></p>
<h2>Modo de preparo</h2>
<p><%= @recipe.prepare_mode %></p>
<h2>Custo</h2>
<p><%= number_to_currency(@recipe.cost) %></p>
</article>

  <%= link_to pluralize(@recipe.comments.size, 'Comentário', 'Comentários'),
  recipe_comments_path(@recipe) %>

</div>

<%= link_to recipes_path, class:'btn btn-link' do %>
  <i class="glyphicon glyphicon-home"> Receitas</i>
<% end %>

<%= link_to edit_recipe_path(@recipe), class:'btn btn-link' do %>
  <i class="glyphicon glyphicon-edit"> Editar</i>
<% end %>

  <%= link_to recipe_path(@recipe), method: :delete, data: { confirm: "Tem certeza que deseja
excluir esta receita?"}, class:'btn btn-link' do %>
    <i class="glyphicon glyphicon-remove-sign"> Deletar</i>
<% end %>

</div>

<br>
```

app/assets/stylesheets/comments.scss

```
$font-size:12px;

*{font-family: 'Slabo 27px', serif;}
a{color: inherit;}
a:hover{color: inherit; text-decoration: none;}

.nova_receita{
  float: right;
}

.imagem_direita{
  float: right;
  margin-right: 30px;
}

.listaReceitasImg img{
  width: 100%;
}

i span{font-family: 'Slabo 27px', serif;}

.small_font_rating{
  font-size: $font-size;
  font-weight: bold;
}

.small_font{
  font-size: $font-size;
}
```

ADICIONANDO AS ALTERAÇÕES NO GIT

```
git status  
git add .  
git status  
git commit -m "Exibindo comentários"  
git push heroku master  
heroku open
```

```
heroku run rails c
```

```
Recipe
```

```
receita = Recipe.find_by(name: 'Estrogonofe de carne')  
receita.comments.create!(name: "Paulo Xavier", rating: 5, comment: "Delicioso")  
receita.comments.create!(name: "Joana Santana", rating: 4, comment: "Muito bom.")  
receita.comments.create!(name: "Augusto Barros", rating: 4, comment: "Gostoso")
```

```
receita = Recipe.find_by(name: 'Salmão ao forno')  
receita.comments.create!(name: "Marta Freitas", rating: 5, comment: "O melhor que já  
comi.")
```

Estrogonofe de carne



Ingredientes

1 kg de carne, 3 cebolas médias, sal e pimenta-do-reino a gosto, 2 latas de creme de leite, 2 colheres (sopa) de ketchup

Calorias

110

Modo de preparo

Em uma panela, adicione o óleo, a carne, a cebola, os tomates, o caldo de carne. Acrescente o ketchup e o champignon. Desligue o fogo e acrescente o creme de leite sem soro. Coloque em uma forma refratária e decore com tempero e batata palha.

Custo

R\$ 50,00

3 Comentários

 [Receitas](#)  [Editar](#)  [Deletar](#)

Receitas fáceis com Receitasnet.

[←](#) [→](#) [C](#) receitasnet-robertopinheiro.herokuapp.com/recipes/1/comments | Pausada R

Receitasnet Receitas

- Paulo Xavier 5 estrelas - 5 minutos atrás
Delicioso
- Joana Santana 4 estrelas - 4 minutos atrás
Muito bom.
- Augusto Barros 4 estrelas - 3 minutos atrás
Gostoso

[Home](#) [Receitas](#)

Receitas fáceis com Receitasnet.

AULA 25 – FORMULÁRIO DE COMENTÁRIOS

app/controllers/comments_controller.rb

```
class CommentsController < ApplicationController
  before_action :set_recipe

  def index
    @comments = @recipe.comments
  end

  def new
    @comment = @recipe.comments.new
  end

  def create
    @comment = @recipe.comments.new(comment_params)
    if @comment.save
      redirect_to recipe_comments_path(@recipe), notice: 'Comentário gravado com sucesso!'
    else
      render :new
    end
  end

  private

  def comment_params
    params.require(:comment).permit(:name, :comment, :rating)
  end

  def set_recipe
    @recipe = Recipe.find(params[:recipe_id])
  end
end
```

config/locales/pt-BR.yml

```
pt-BR:  
activerecord:  
  models:  
    recipe: Receita  
      comment: Comentário  
    attributes:  
      recipe:  
        name: Nome da Receita  
        stuff: Ingredientes  
        calories: Calorias  
        kind: Tipo  
        prepare_mode: Modo de preparo  
        cost: Custo  
        portion: Porção  
        duration: Tempo de preparo  
        poster: Poster  
      comment:  
        name: Nome  
        rating: Estrelas  
        comment: Comentário  
  errors:  
    messages:  
      record_invalid: 'A validação falhou: %{errors}'  
      restrict_dependent_destroy:  
        has_one: Não é possível excluir o registro pois existe um %{record} dependente  
        has_many: Não é possível excluir o registro pois existem %{record} dependentes  
  date:  
    abbr_day_names:  
      - dom  
      - seg  
      - ter  
      - qua  
      - qui  
      - sex  
      - sáb  
    abbr_month_names:  
      -  
      - jan  
      - fev  
      - mar  
      - abr  
      - mai  
      - jun  
      - jul  
      - ago  
      - set  
      - out  
      - nov
```

```
- dez
day_names:
- domingo
- segunda-feira
- terça-feira
- quarta-feira
- quinta-feira
- sexta-feira
- sábado
formats:
default: "%d/%m/%Y"
long: "%d de %B de %Y"
short: "%d de %B"
month_names:
-
- janeiro
- fevereiro
- março
- abril
- maio
- junho
- julho
- agosto
- setembro
- outubro
- novembro
- dezembro
order:
- :day
- :month
- :year
datetime:
distance_in_words:
about_x_hours:
one: aproximadamente 1 hora
other: aproximadamente %{count} horas
about_x_months:
one: aproximadamente 1 mês
other: aproximadamente %{count} meses
about_x_years:
one: aproximadamente 1 ano
other: aproximadamente %{count} anos
almost_x_years:
one: quase 1 ano
other: quase %{count} anos
half_a_minute: meio minuto
less_than_x_seconds:
one: menos de 1 segundo
other: menos de %{count} segundos
less_than_x_minutes:
one: menos de um minuto
other: menos de %{count} minutos
```

```
over_x_years:  
    one: mais de 1 ano  
    other: mais de %{count} anos  
x_seconds:  
    one: 1 segundo  
    other: "%{count} segundos"  
x_minutes:  
    one: 1 minuto  
    other: "%{count} minutos"  
x_days:  
    one: 1 dia  
    other: "%{count} dias"  
x_months:  
    one: 1 mês  
    other: "%{count} meses"  
x_years:  
    one: 1 ano  
    other: "%{count} anos"  
prompts:  
    second: Segundo  
    minute: Minuto  
    hour: Hora  
    day: Dia  
    month: Mês  
    year: Ano  
errors:  
    format: "%{attribute} %{message}"  
    messages:  
        accepted: deve ser aceito  
        blank: não pode ficar em branco  
        confirmation: não é igual a %{attribute}  
        empty: não pode ficar vazio  
        equal_to: deve ser igual a %{count}  
        even: deve ser par  
        exclusion: não está disponível  
        greater_than: deve ser maior que %{count}  
        greater_than_or_equal_to: deve ser maior ou igual a %{count}  
        inclusion: não está incluído na lista  
        invalid: não é válido  
        less_than: deve ser menor que %{count}  
        less_than_or_equal_to: deve ser menor ou igual a %{count}  
        model_invalid: 'A validação falhou: %{errors}'  
        not_a_number: não é um número  
        not_an_integer: não é um número inteiro  
        odd: deve ser ímpar  
        other_than: deve ser diferente de %{count}  
        present: deve ficar em branco  
        required: é obrigatório(a)  
        taken: já está em uso  
        too_long:  
            one: 'é muito longo (máximo: 1 caracter)'  
            other: 'é muito longo (máximo: %{count} caracteres)'  
    
```

```
too_short:
  one: 'é muito curto (mínimo: 1 caracter)'
  other: 'é muito curto (mínimo: %{count} caracteres)'

wrong_length:
  one: não possui o tamanho esperado (1 caracter)
  other: não possui o tamanho esperado (%{count} caracteres)

template:
  body: 'Por favor, verifique o(s) seguinte(s) campo(s):'
  header:
    one: 'Não foi possível gravar %{model}: 1 erro'
    other: 'Não foi possível gravar %{model}: %{count} erros'

helpers:
  select:
    prompt: Por favor selecione
  submit:
    create: Criar %{model}
    submit: Salvar %{model}
    update: Atualizar %{model}

number:
  currency:
    format:
      delimiter: "."
      format: "%u %n"
      precision: 2
      separator: ","
      significant: false
      strip_insignificant_zeros: false
      unit: R$ 

    format:
      delimiter: "."
      precision: 3
      separator: ","
      significant: false
      strip_insignificant_zeros: false

  human:
    decimal_units:
      format: "%n %u"
      units:
        billion:
          one: bilhão
          other: bilhões
        million:
          one: milhão
          other: milhões
        quadrillion:
          one: quatrilhão
          other: quatrilhões
        thousand: mil
        trillion:
          one: trilhão
          other: trilhões
        unit: "
```

```
format:  
  delimiter: ""  
  precision: 3  
  significant: true  
  strip_insignificant_zeros: true  
storage_units:  
  format: "%n %u"  
  units:  
    byte:  
      one: Byte  
      other: Bytes  
    eb: EB  
    gb: GB  
    kb: KB  
    mb: MB  
    pb: PB  
    tb: TB  
percentage:  
  format:  
    delimiter: "."  
    format: "%n%"  
precision:  
  format:  
    delimiter: "."  
support:  
array:  
  last_word_connector: " e "  
  two_words_connector: " e "  
  words_connector: ", "  
time:  
am: ""  
formats:  
  default: "%a, %d de %B de %Y, %H:%M:%S %z"  
  long: "%d de %B de %Y, %H:%M"  
  short: "%d de %B, %H:%M"  
pm: ""
```

app/views/comments/index.html.erb

```
<div class="container">

<% if @recipe.comments.size > 0 %>
<div class="jumbotron">
<ul>
<% @comments.each do |comment| %>
<li>
<p>
<%= comment.name %> <small class="small_font_rating"><%= pluralize(comment.rating,
'estrela', 'estrelas') %></small>
<small class="small_font"> - <%= time_ago_in_words(comment.created_at) %>
atrás</small>
</p>
<p>
<%= comment.comment %>
</p>
</li>
<% end %>
</ul>
</div>
<% else %>
<div class="jumbotron">
<h3 class="text-center">Não existem comentários!</h3>
</div>
<% end %>

<%= link_to recipes_path, class:'btn btn-link' do %>
<i class="glyphicon glyphicon-home"> Receitas</i>
<% end %>

</div>

<br>
```

app/views/comments/new.html.erb

```
<div class="container">
<h2>Criando novo Comentário</h2>
<%= render 'form' %>
</div>
```

app/views/comments/_form.html.erb

```
<%= form_for(@comment) do |f| %>
<%= render 'layouts/errors', object: @comment %>
<p class="form-group">
<%= f.label :Nome %><br/>
<%= f.text_field :name, autofocus: true, class:"form-control" %>
</p>
<p class="form-group">
<%= f.label :Estrelas %><br/>
<%= f.select :rating, Comment::RATING , prompt: 'Escolha a quantidade de estrelas', class:"form-control"%>
</p>
<p class="form-group">
<%= f.label :Comentário %><br/>
<%= f.text_area :comment, placeholder:"Escreva seu comentário ..." ,class:"form-control" %>
</p>
<p>
<%= f.submit class:'btn btn-primary' %>
<%= link_to 'Cancelar', root_path , class: 'btn btn-outline-danger'%>
</p>
<% end %>

<br>
```

app/views/recipes/show.html.erb

```
<div class="container">
<div class="jumbotron">
<article>
<head>
  <h1><%= @recipe.name %></h1>
  <%= imagem(@recipe) %>
</head>
<h2>Ingredientes</h2>
<p><%= @recipe.stuff %></p>
<h2>Calorias</h2>
<p><%= @recipe.calories %></p>
<h2>Modo de preparo</h2>
<p><%= @recipe.prepare_mode %></p>
<h2>Custo</h2>
<p><%= number_to_currency(@recipe.cost) %></p>
</article>

<%= link_to pluralize(@recipe.comments.size, 'Comentário', 'Comentários'),
  recipe_comments_path(@recipe) %>
  <%= link_to 'Novo comentário', new_recipe_comment_path(@recipe), class: 'btn btn-primary' %>

</div>

<%= link_to recipes_path, class:'btn btn-link' do %>
  <i class="glyphicon glyphicon-home"> Receitas</i>
<% end %>

<%= link_to edit_recipe_path(@recipe), class:'btn btn-link' do %>
  <i class="glyphicon glyphicon-edit"> Editar</i>
<% end %>

<%= link_to recipe_path(@recipe), method: :delete, data: { confirm: "Tem certeza que deseja
excluir esta receita?"}, class:'btn btn-link' do %>
  <i class="glyphicon glyphicon-remove-sign"> Deletar</i>
<% end %>

</div>

<br>
```

app/views/recipes/_form.html.erb

```
<%= form_for(@recipe) do |f| %>

<%= render 'layouts/errors', object: @recipe %>



<%= f.label :Nome %><br>
<%= f.text_field :name, class: 'form-control' %>



<%= f.label :Ingredientes %><br>
<%= f.text_area :stuff, cols: 40, rows: 5, class: 'form-control' %>



<%= f.label :Tipo %><br>
<%= f.select :kind, Recipe::KIND, prompt: 'Escolha uma opção', class: 'form-control' %>



<%= f.label :Calorias %><br>
<%= f.number_field :calories, class: 'form-control' %>



<%= f.label :Modo_de_Preparo %><br>
<%= f.text_area :prepare_mode, cols: 40, rows: 5, class: 'form-control' %>



<%= f.label :Tempo_de_Preparo %><br>
<%= f.time_select :duration, class: 'form-control' %>



<%= f.label :Porções %><br>
<%= f.text_field :portion, class: 'form-control' %>



<%= f.label :Custo %><br>
<%= f.number_field :cost, class: 'form-control' %>



<%= f.label :Poster %><br>
<%= f.text_field :poster, placeholder: ".gif, .jpeg, .png", class: 'form-control' %>


```

```
<p>
  <%= f.submit class:"btn btn-primary" %>
  <%= link_to "Cancelar", root_path, class:"btn btn-default" %>
</p>
<% end %>
```

app/views/layouts/application.erb

```
<% if object.errors.any? %>
<div class="alert alert-danger" role="alert">
<ul>
  <% object.errors.full_messages.each do |message| %>
    <li><%= message %></li>
  <% end %>
</ul>
</div>
<% end %>
```

ADICIONANDO AS ALTERAÇÕES NO GIT

```
git status
git add .
git status
git commit -m "Criando formulário de comentários"
git push heroku master
heroku open
```

The screenshot shows a web page with a header containing a logo and the text 'Receitasnet' on the left, and 'Receitas' on the right. Below the header is a title 'Criando novo Comentário'. The form has three fields: 'Nome' with the value 'Roberto Pinheiro', 'Estrelas' with a dropdown menu showing '4', and 'Comentário' with the text 'Leve e saborosa!'. At the bottom are two buttons: 'Criar Comentário' (in blue) and 'Cancelar'.

Criando novo Comentário

Nome

Roberto Pinheiro

Estrelas

4

Comentário

Leve e saborosa!

Criar Comentário Cancelar

Comentário gravado com sucesso! ×

• Roberto Pinheiro 4 estrelas - menos de um minuto atrás

Leve e saborosa!

 Receitas

Receitas fáceis com Receitasnet.