

Лекция 7. Шаблоны. Model-View-Controller

- Rack - промежуточный слой
- Sinatra DSL
- Принципы MVC и Ruby on Rails

МГТУ им. Н.Э. Баумана, доц. каф. ИУ-6,
к.т.н. Самарев Роман Станиславович

samarev@acm.org

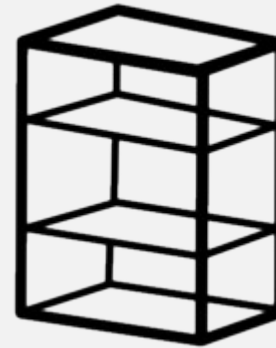


Rack: a Ruby Webserver Interface



15.10.2019

- Christian Neukirchen
- Rack 0.1: March 3rd, 2007
- Rack 1.4.1: January 23rd, 2012
- Rack 1.5.2: February 7th, 2013
- Rack 2.0.1: June 30th, 2016
- Rack 2.0.7: April 02th, 2019
- `gem install rack`
- <http://rack.github.com/>

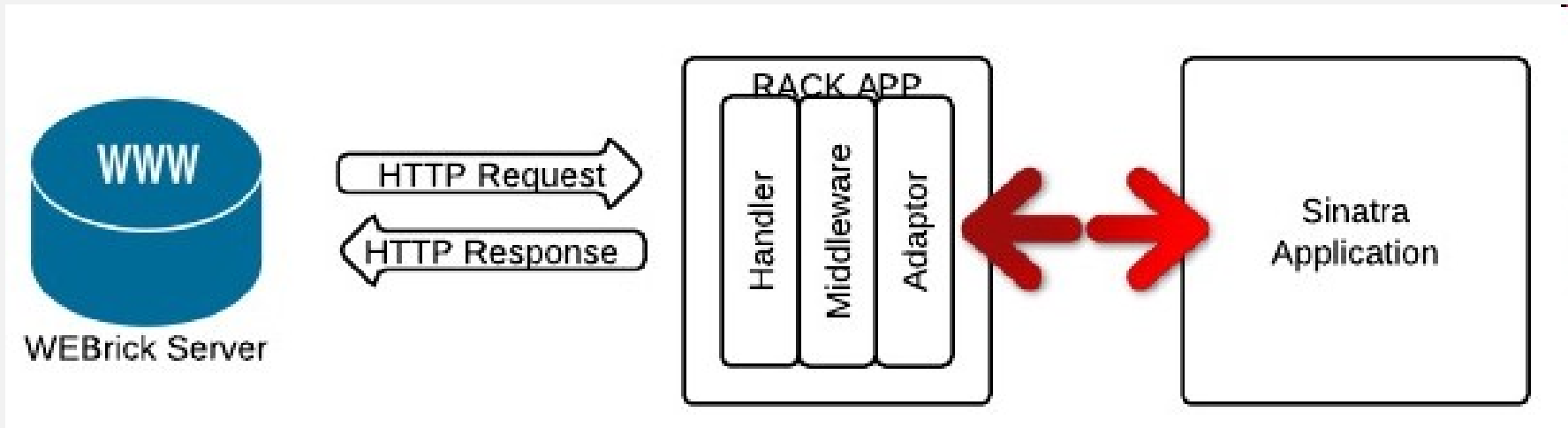


rack
powers web applications

Rack Middleware



15.10.2019



- <http://rubylearning.com/blog/2013/04/02/whats-rack/>

Rack

Supported web servers



15.10.2019

- Веб-серверы, поддерживающие взаимодействие с Rack:
 - Mongrel
 - EventedMongrel
 - SwiftliedMongrel
 - **WEBrick**
 - FCGI
 - CGI
 - SCGI
 - LiteSpeed
 - **Thin**
- Веб-серверы, содержащие Rack в комплекте поставки:
 - Ebb
 - Fuzed
 - Glassfish v3
 - Phusion Passenger (mod_rack для Apache и nginx)
 - **Puma**
 - Rainbows!
 - **Unicorn**
 - Zbatory

Rack

Supported web frameworks



15.10.2019

- Camping
- Coset
- Halcyon
- **Hanami**
- Mack
- Maveric
- Merb
- Racktools::SimpleApplication
- Ramaze
- **Ruby on Rails**
- Rum
- **Sinatra**
- Sin
- Vintage
- Waves
- Wee
- ...



- Минимальный модульный интерфейс
- Rack-приложение – объект Ruby
- На входе: `environment`
- Результат: `status, headers, body`.

```
run Proc.new {|env|  
  [200, {"Content-Type" => "text/html"}, "Hello Rack!"]}
```

Rack

HTTP-переменные



15.10.2019

- REQUEST_METHOD
- SCRIPT_NAME
- PATH_INFO
- QUERY_STRING
- SERVER_NAME, SERVER_PORT, SCRIPT_NAME PATH_INFO
- HTTP_HOST
- HTTP_ Variables
- <http://rack.rubyforge.org/doc/SPEC.html>

Rack

специфические переменные



15.10.2019

<code>rack.version</code>	Версия Rack
<code>rack.url_scheme</code>	http или https
<code>rack.input</code>	Входной поток
<code>rack.errors</code>	Поток вывода ошибок
<code>rack.multithread</code>	true если приложение многопоточное (общий процесс)
<code>rack.multiprocess</code>	true если приложение запускается в виде параллельных процессов.
<code>rack.run_once</code>	true если приложение завершается после выполнения запроса(режим CGI).



- Минимальная программа **config.ru**

```
infinity = Proc.new do |env|  
  [ 200,  
    {"Content-Type" => "text/html"},  
    "<html>" + env.to_s + "</html>" ]  
end  
run infinity
```

- Запуск: **rackup config.ru** или **thin start**

Rack

Фильтрация адреса (map)



15.10.2019

```
infinity =  
Proc.new {|env| [200, {'Content-Type' => 'text/html'}, env.inspect]}
```

builder = Rack::Builder.new do
 use Rack::CommonLogger

map '/' do
 run infinity
 end

map '/version' do
 run Proc.new {|env| [200, {'Content-Type' => 'text/html'}, 'infinity 0.1'] }
 end
end

Rack::Handler::Mongrel.run builder, :Port => 9292

Rack

Вложенные блоки map



15.10.2019

```
infinity = Proc.new {|env| [200, {'Content-Type' => 'text/html'}, env.inspect]}
```

```
use Rack::CommonLogger
```

```
map '/' do
  run infinity
end
```

```
map '/version' do
  map '/' do
    run Proc.new {|env| [200, {'Content-Type' => 'text/html'}, 'infinity 0.1'] }
  end
end
```

```
map '/last' do
  run Proc.new {|env| [200, {'Content-Type' => 'text/html'}, 'infinity beta 0.0'] }
end
end
```



```
infinity = Proc.new { |env|  
  date = DateTime.now.strftime("%F")  
  body = File.read('index.html').gsub(/%DATE%/, date)  
  [ 200,  
    {'Content-Type'=>'text/html',  
      'Content-Length' => body.length.to_s,  
      'Cache-Control' => 'public, max-age=3600' },  
    body  
  ]  
}
```

run infinity

Rack

дополнительные материалы



15.10.2019

- <http://rubylearning.com/blog/2010/09/21/writing-modular-web-applications-with-rack/>

Sinatra

DSL for quickly creating web applications



15.10.2019

- Blake Mizerany, Konstantin Haase, Ryan Tomayko, Simon Rozet
- 0.0.1 September 09, 2007
- 1.3.3 August 18, 2012
- 1.4.3 June 7, 2013
- 1.4.7 January 24, 2016
- 2.0.0 May 07, 2017
- 2.0.7 August 22, 2019
- Легковесный фреймворк
- DSL - domain-specific language
- Обеспечивает:
«**быстрое создание веб-приложений с минимальными усилиями**»
- <http://www.sinatrarb.com/>



Sinatra

Пример веб-приложения



15.10.2019

```
#!/usr/bin/env ruby  
require 'sinatra'
```

```
get '/' do      # обработчик для адреса /  
  redirect to('/hello/World')  
end
```

```
get '/hello/:name' do # обработчик адреса /hello/ИМЯ  
  "Hello #{params[:name]}!"  
end
```

- <http://www.sinatrarb.com/intro>



get '/' do

.. show something ..

end

post '/' do

.. create something ..

end

put '/' do

.. replace something ..

end

patch '/' do

.. modify something ..

end

delete '/' do

.. annihilate something ..

end

options '/' do

.. appease something ..

end

Sinatra


Маршруты




15.10.2019

- Подстановка переменной через params

```
get '/hello/:name' do
  # matches "GET /hello/foo" and "GET /hello/bar"
  # params[:name] is 'foo' or 'bar'
  "Hello #{params[:name]}!"
end
```


- Подстановка переменной через блок

```
get '/hello/:name' do |n|
  "Hello #{n}!"
end
```



Sinatra

УСЛОВИЯ



15.10.2019

```
get '/', :host_name => /^admin\.\/ do
  "Admin Area, Access denied!"
end
```

```
get '/', :provides => 'html' do
  haml :index
end
```

```
get '/', :provides => ['rss', 'atom', 'xml'] do
  builder :feed
end
```

Sinatra

Static files



15.10.2019

Разрешение доступа к статическим файлам в указанной директории

```
set :public_folder, File.dirname(__FILE__) + '/static'
```

- `File.dirname(__FILE__)` - получить директорию с текущим скриптом Ruby

Sinatra

Views / Templates



15.10.2019

```
get '/' do
  erb :index
end
```

Шаблоны
размещают в
директории

```
get '/' do
  code = "<%= Time.now %>"
  erb code
end
```

./views

```
get '/' do
  haml :index, :format => :html5
end
```

Sinatra

Фильтры



15.10.2019

before do

 @note = 'Hi!'

 request.path_info = '/foo/bar/baz'

end

get '/foo/*' do

 @note + #=> 'Hi!'

 params[:splat] #=> 'bar/baz'

end



```
get '/' do  
  send_file 'foo.png'  
end
```

```
send_file 'foo.png', :type => :jpg
```



```
get '/foo' do
  "<a href='/bar'>do something</a>"
end
```

```
get '/bar' do
  do_something
  redirect back
end
```

```
redirect to('/bar'), 303
redirect 'http://google.com', 'wrong place, buddy'
```

Дополнительные материалы



15.10.2019

- <http://www.sinatrarb.com/intro>
- <https://github.com/sinatra/sinatra-book>
- <http://www.slideshare.net/benschwarz/sinatra-rack-and-middleware-1509268>



Model

View

Controller



- Набор данных
- Правила применения
- Поддержание логической целостности
- Пример:
 - Не предоставлять скидку на заказы дешевле 1000 руб

View (представление)



15.10.2019

- Формирование интерфейса пользователя на основе модели
- Ввод данных различными способами
- Представление данных с различными целями
(редактирование или просмотр, и пр.)
- **Никогда не занимается обработкой!**

Controller (Контроллер)



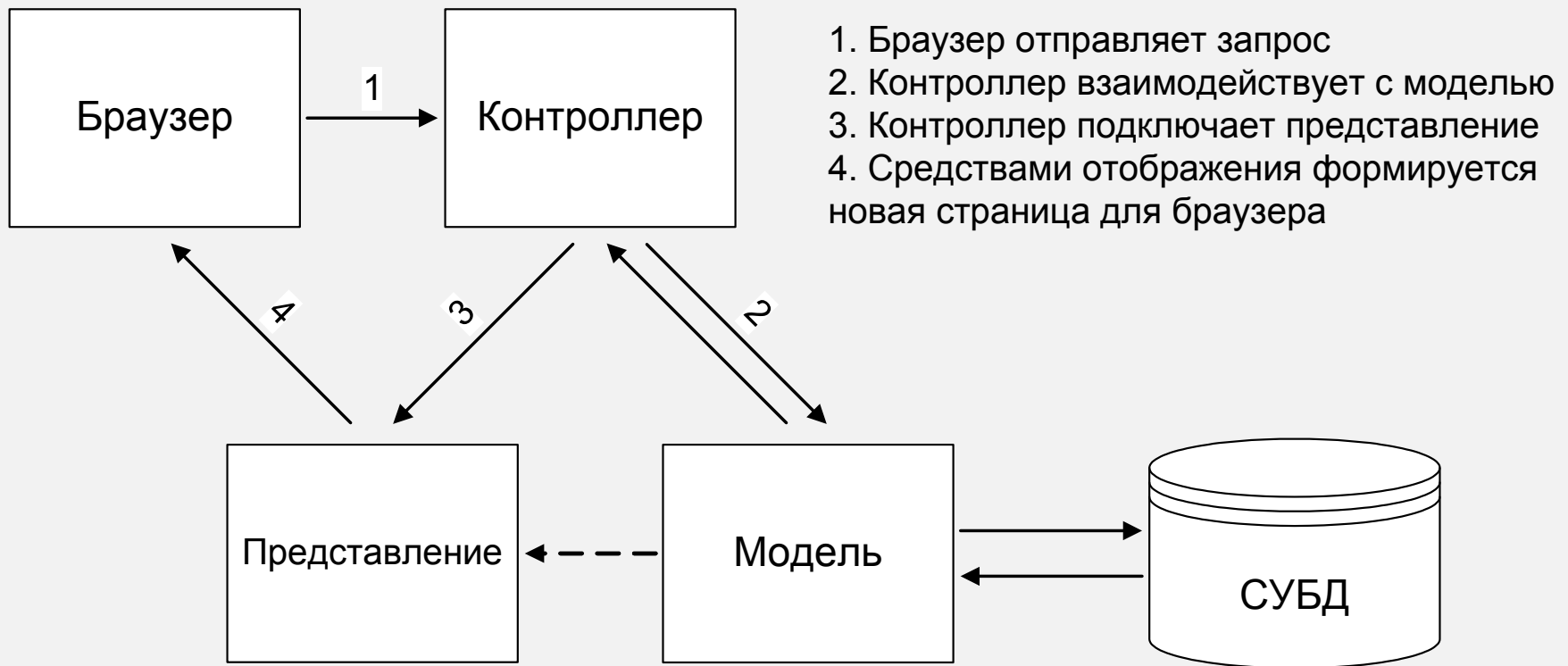
15.10.2019

- Связывают отдельные элементы приложения
- Получают внешние события
- Взаимодействуют с моделями
- Активируют представления

Схема разделения управления Model-View-Controller



15.10.2019



Ruby on Rails



15.10.2019

- David Heinemeier Hansson
- December 13, 2005 – 1.0
- December 7, 2007 – 2.0
- August 29, 2010 – 3.0
- June 25, 2013 – 4.0
- August 10, 2016 – 5.0.0.1
- September 08, 2017 – 5.1.4
- March 28, 2019 – 5.2.3
- August 16, 2019 – 6.0.0
- <http://rubyonrails.org/>



Состав Ruby on Rails



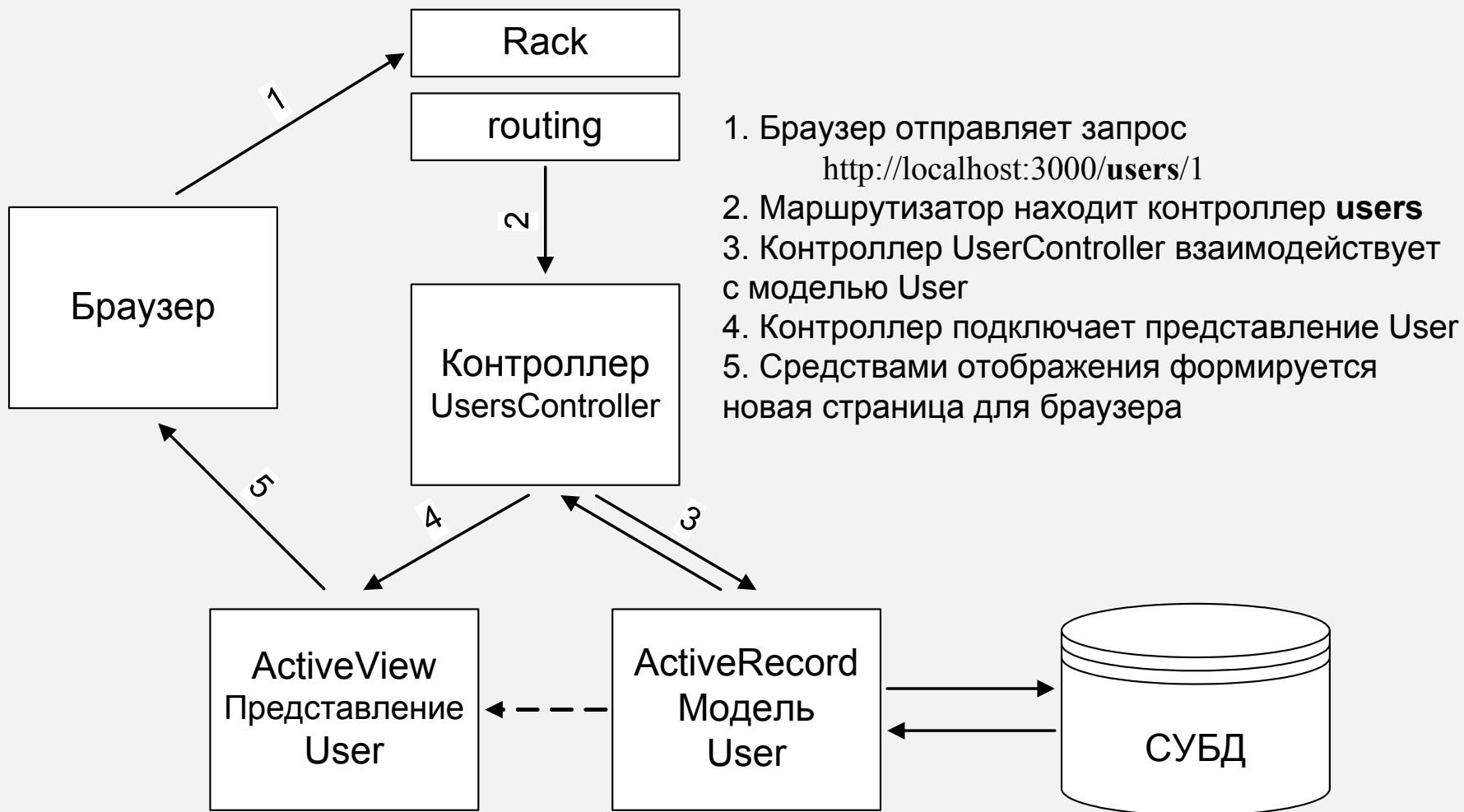
15.10.2019

- ActiveRecord
- ActiveResource
- ActionPack
- ActiveSupport
- ActionMailer
- ...

Архитектура Ruby on Rails



15.10.2019



Генерация приложения



15.10.2019

- Предварительная установка rails
`gem install rails`
- Создание нового приложения *test_app*
`rails new test_app`

Структура Rails приложения



15.10.2019

- шаблон краткой информации о приложении
README.rdoc
- Файл системы сборки Rake (аналог make для Ruby)
Rakefile
- Вспомогательные файлы
config.ru
.gitignore
- Файл необходимых gem-пакетов.
Gemfile

Структура Rails приложения



15.10.2019

- основной каркас: контроллеры, представления, модели, «хелперы», шаблоны типового отображения:

app

app/assets/images/rails.png

app/assets/javascripts/application.js

app/assets/stylesheets/application.css

app/controllers/application_controller.rb

app/helpers/application_helper.rb

app/mailers

app/models

app/views/layouts/application.html.erb

app/mailers/.gitkeep

app/models/.gitkeep

Структура Rails приложения



15.10.2019

- конфигурационная часть - маршрутизацию, настройки, локали:
config
config/routes.rb
config/application.rb
config/environment.rb
config/environments
config/environments/development.rb
config/environments/production.rb
config/environments/test.rb
config/initializers
config/initializers/backtrace_silencers.rb
config/initializers/inflections.rb
config/initializers/mime_types.rb
config/initializers/secret_token.rb
config/initializers/session_store.rb
config/initializers/wrap_parameters.rb
config/locales
config/locales/en.yml
config/boot.rb
config/database.yml

Структура Rails приложения



15.10.2019

- Базы данных:
db
db/seeds.rb
- Директория для будущей документации:
doc
doc/README_FOR_APP
- Директории для библиотек функций:
lib
lib/tasks
lib/tasks/.gitkeep
lib/assets
lib/assets/.gitkeep
- Журналы выполнения приложения:
log
log/.gitkeep

Структура Rails приложения



15.10.2019

- Директория для общедоступных статических файлов:
 - public
 - public/404.html
 - public/422.html
 - public/500.html
 - public/favicon.ico
 - public/index.html
 - public/robots.txt
- Директория служебных скриптов Rails:
 - script
 - script/rails

Структура Rails приложения



15.10.2019

- Директория тестов приложения:
 - test/fixtures
 - test/fixtures/.gitkeep
 - test/functional
 - test/functional/.gitkeep
 - test/integration
 - test/integration/.gitkeep
 - test/unit
 - test/unit/.gitkeep
 - test/performance/browsing_test.rb
 - test/test_helper.rb

Структура Rails приложения



15.10.2019

- Временная директория для служебных целей.
Она же для хранения кэша:
 tmp/cache
 tmp/cache/assets
- Директории для размещения дополнительных модулей:
 vendor/assets/javascripts
 vendor/assets/javascripts/.gitkeep
 vendor/assets/stylesheets
 vendor/assets/stylesheets/.gitkeep
 vendor/plugins
 vendor/plugins/.gitkeep



- **rails server**
 - **WEBrick** (отладочный веб-сервер) для Rails ≤ 5.1
 - **puma** для Rails > 5.1
- **thin start**
 - **Thin** (минимальный веб-сервер)

Rails Scaffold - генератор «строительных лесов»



15.10.2019

- rails generate scaffold User **name:string email:string**



15.10.2019

Генерация каркаса

- Создание спецификации базы данных:
invoke active_record
create db/migrate/20170817160323_create_users.rb
- Создание эквивалентной модели:
create app/models/user.rb
- Создание Unit-тестов приложения:
invoke test_unit
create test/unit/user_test.rb
create test/fixtures/users.yml
- Прописывание нового маршрута:
invoke resource_route
route resources :users

Генерация каркаса



15.10.2019

- Создание нового контроллера:
 `invoke scaffold_controller`
 `create app/controllers/users_controller.rb`
- Генерация представлений:
 `invoke erb`
 `create app/views/users`
 `create app/views/users/index.html.erb`
 `create app/views/users/edit.html.erb`
 `create app/views/users/show.html.erb`
 `create app/views/users/new.html.erb`
 `create app/views/users/_form.html.erb`

Генерация каркаса



15.10.2019

- Создание функционального теста:
 `invoke test_unit`
 `create test/functional/users_controller_test.rb`
- Создание помощника:
 `invoke helper`
 `create app/helpers/users_helper.rb`
- Создание теста для него:
 `invoke test_unit`
 `create test/unit/helpers/users_helper_test.rb`



- Создание «полезностей» (Assets):
 invoke assets
 invoke coffee
 create app/assets/javascripts/users.js.coffee
 invoke scss
 create app/assets/stylesheets/users.css.scss
 invoke scss
 create app/assets/stylesheets/scaffolds.css.scss



- `config/routes.rb`
 - Результат scaffold-генератора:

`resources :users`

4 маршрута (в концепции CRUD - Create Read Update Destroy)

7 действий всего: `index`, `show`, `new`, `create`, `edit`, `update`, `destroy`

Контроллер UserController



15.10.2019

- app/controllers/user_controller.rb

```
class UserController < ApplicationController
```

```
  # GET /users
```

```
  # GET /users.json
```

```
  def index
```

```
    @users = User.all
```

```
    respond_to do |format|
```

```
      format.html # index.html.erb
```

```
      format.json { render json: @users }
```

```
    end
```

```
  end
```

```
  # GET /users/1
```

```
  # GET /users/1.json
```

```
  def show
```

```
    @user = User.find(params[:id])
```

```
    respond_to do |format|
```

```
      format.html # show.html.erb
```

```
      format.json { render json: @user }
```

```
    end
```

```
  end
```

```
...
```


Модель User



15.10.2019

- app/models/user.rb

```
class User < ActiveRecord::Base
  attr_accessible :email, :name
end
```

Представления

Общий шаблон отображения



15.10.2019

- app/views/layouts/application.html.erb

```
<!DOCTYPE html>
<html>
<head>
  <title>TestApp</title>
  <%= stylesheet_link_tag "application", :media => "all" %>
  <%= javascript_include_tag "application" %>
  <%= csrf_meta_tags %>
</head>
<body>
<%= yield %>
</body>
</html>
```

Представления

Шаблон users/index



15.10.2019

- app/views/users/index.html.erb

```
<h1>Listing users</h1>
<table>
  <tr>
    <th>Name</th>
    <th>Email</th>
    <th></th>
    <th></th>
    <th></th>
  </tr>
  <% @users.each do |user| %>
    <tr>
      <td><%= user.name %></td>
      <td><%= user.email %></td>
      <td><%= link_to 'Show', user %></td>
      <td><%= link_to 'Edit', edit_user_path(user) %></td>
      <td><%= link_to 'Destroy', user, method: :delete, data: { confirm: 'Are you sure?' } %></td>
    </tr>
  <% end %>
</table>
<br />
<%= link_to 'New User', new_user_path %>
```

Генерация контроллера без scaffold-генератора



15.10.2019

- rails generate controller Calc input view

Контроллер Calc



15.10.2019

- app/controllers/calc_controller.rb

```
class CalcController < ApplicationController
  def input
  end
  def view
    v1 = params[:v1].to_i
    v2 = params[:v2].to_i
    @result = case params[:op]
      when "+" then v1 + v2;
      when "-" then v1 - v2;
      when "*" then v1 * v2;
      when "/" then v1 / v2;
      else "Unknown!"
    end
  end
end
```



15.10.2019

Представление «Input»

- app/views/input.html.erb

```
<h1>Calc#input</h1>
<p>Find me in app/views/calc/input.html.erb</p>

<%= form_tag("/calc/view", :method => "get") do %>
  <%= label_tag("Value 1:") %>
  <%= text_field_tag(:v1) %> <br/>
  <%= label_tag("Value 2") %>
  <%= text_field_tag(:v2) %> <br/>

  <%= ['+', '-', '*', '/'].each do |op| %>
    <%= label_tag(op) %>
    <%= radio_button_tag(:op, op) %><br/>
  <br/>
  <%= end %>

  <%= submit_tag("Calc result") %>
<%= end %>
```

Представление «View»



15.10.2019

- `app/views/view.html.erb`

```
<h1>Calc#view</h1>
```

```
<p>Find me in app/views/calc/view.html.erb</p>
```

```
<p id="result"><%= @result %></p>
```

```
<%= link_to "Repeat calculation", :calc_input %>
```

Результат



15.10.2019

Calc#input

Find me in app/views/calc/input.html.erb

Value 1:

Value 2:

+ ☐

- ☐

* ☒

/ ☐

Calc result

Calc#view

Find me in app/views/calc/view.html.erb

20

[Repeat calculation](#)



- <http://rubyonrails.org/>
- <http://guides.rubyonrails.org/>
- <http://rusrails.ru/>
- <https://www.railstutorial.org/book>
- Sam Ruby, Dave Thomas, David Heinemeier Hansson. Agile Web Development with Rails. Third Edition. The Pragmatic Bookshelf. Raleigh, North Carolina Dallas, Texas. 2009
- Obie Fernandez. The Rails 5 Way. Addison-Wesley. 2017
- Гибкая разработка веб-приложений в среде Rails. 4-е издание Сэм Руби, Дэйв Томас, Дэвид Хэнссон. Серия: Для профессионалов.- Питер: 2013.- 464 стр.