

Ruby on Rails

Daniel Mrózek, @mrazicz

na začatek

- nepoužívám diakritiku, nehledejte ji ani tady
- pracuji stejně jak tady Honza nebo Tom ve firmě weps.cz
- dvě přednášky o Railsích, dnes a za dva týdny
- co budete potřebovat (zejména na cviku):
 - textový editor, terminal, ideálně Linux nebo Unix
 - ruby \geq 1.9.2 a gemy: bundler, rails \geq 4.0.0; připojení na net
 - základní znalosti git-u a příkazové řádky
 - a hlavně pevné nervy abyste vydrželi mě poslouchat :)

trocha historie

- 1993: Yukihiro Matsumoto (Matz) zacina pracovat na ruby
- 1995: prvnι verze ruby
- 2004: aplikace Basecamp od 37signals - z toho pozdeji D. H. Hansson (@dhh) “vyextrahoval” RoR
- 2005: Ruby on Rails 1.0
- 2013: RoR 4.0

bundler

- pro spravu gem-u a jejich zavislosti v ramci projektu
- touch Gemfile

```
echo `source "http://rubygems.org"` >> Gemfile
```

```
echo `gem "nokogiri", "~> 1.6.0"` >> Gemfile
```

```
bundle install --path /path/to/dir
```

```
bundle exec irb
```

```
>> require 'nokogiri'
```

```
=> true
```

git

- viz. prednaska Toma Hrabala (Lecture 03)
- dnes nam bude staci: git init, add, commit
- `git add -i` velmi uzitecny command
- `.gitignore`
- `.gitkeep` popr. `.keep`
 - s gitem nic spolecneho
 - pouzivame pro zachovani prazdnych adresaru v repositari

rbenv, RVM

- pro vice verzi ruby na jednom stroji
- pro development, do produkce nedoporučuje
- jednoduché prepínání z jedné verze do druhé
- poměrně jednoduchá instalace

voliteľne

- slim-lang.com - zkúsite a html už nebudete nikdy chtiť
- coffeescript.org - .js sa muže jít schovať
- haml.info - alternativa ke slim-u
- sass, scss - nahrada css, promenenne, zanorovani, mixiny apod.

kdy (ne)pouzit railsy

- Jednoduchá webová prezentace? ne
- Jednoduché API napr. nad DB? asi ne
- Blog? asi ano
- Rozsáhlejší prezentace? ano
- Eshop, CMS, větší APP? ano
- Rozsáhlé API? ano

Rails - konecne

- `gem install rails`
 - `rails new myapp -d mysql`
 - `cd myapp`
 - `bundle install --path .bundle`
 - `echo 'config/*.yml' >> .gitignore`
 - `echo 'Gemfile.lock' >> .gitignore`
 - `cp config/database.yml config/database.yml.example`
 - `vim config/database.yml # setup db connection`
 - `git init && git add .`
 - `git commit -m '[INIT]ial commit'`
 - `bundle exec rails s`
- # open browser and go to <http://localhost:3000>

struktura

```
% tree -L 1
.
├── Gemfile
├── Gemfile.lock
├── README.rdoc
├── Rakefile
├── app
├── bin
├── config
├── config.ru
├── db
├── lib
├── log
├── public
├── test
├── tmp
└── vendor

10 directories, 5 files
%
```

```
| % tree app
|app
|├── assets
|│   ├── images
|│   ├── javascripts
|│   │   └── application.js
|│   └── stylesheets
|│       └── application.css
|├── controllers
|│   ├── application_controller.rb
|│   └── concerns
|├── helpers
|│   └── application_helper.rb
|├── mailers
|├── models
|│   └── concerns
|├── views
|│   ├── layouts
|│   └── application.html.erb
|
|12 directories, 5 files
| %
```

```
| % tree config
|config
|├── application.rb
|├── boot.rb
|├── database.yml
|├── environment.rb
|├── environments
|│   ├── development.rb
|│   ├── production.rb
|│   └── test.rb
|├── initializers
|│   ├── backtrace_silencers.rb
|│   ├── filter_parameter_logging.rb
|│   ├── inflections.rb
|│   ├── mime_types.rb
|│   ├── secret_token.rb
|│   ├── session_store.rb
|│   └── wrap_parameters.rb
|├── locales
|│   └── en.yml
|└── routes.rb
|
|3 directories, 16 files
```

```
| % tree test tmp/myapp
|test
|├── controllers
|├── fixtures
|├── helpers
|├── integration
|├── mailers
|├── models
|└── test_helper.rb
|
|6 directories, 1 file
| % tree vendor
|vendor
|├── assets
|│   ├── javascripts
|│   └── stylesheets
|
|3 directories, 0 files
| % █ tmp/myapp
```

config/routes

- zde urcujeme routovani
- viz. ukazka

app/assets

- zde patri styly, javascript, obrazky atd.
- v railsech vetsinou budeme pouzivat:
 - misto css: scss nebo sass
 - misto js: coffeescript
- manifesty (js, css) - co vse se ma “includnout”

app/controllers

- MVC
- ActionController::Base
- ApplicationController < ActionController::Base
 - základni controller, zde si budeme davat spolecne metody pro zbytek controlleru
- ridici logika aplikace: najdi objekt, uloz objekt, vykresli sablonu, vyfiltruj parametry atd.

app/helpers

- pomocne metody pro views
- vetsinou je pojmenovavame stejne jako controllery
 - UsersController => UsersHelper

app/mailers

- Obsluha emailu
- `UserMailer < ActionMailer::Base`
- pohledy ukladame normalne do app/views
- `UserMailer.welcome_email(@user).deliver`

app/models

- **M**_{VC}
- datove modely
- zaznamy z databaze mapujeme na modely - objekty
- `class User < ActiveRecord::Base`

app/views

- **MVC**
- `ActionView::Base`
- zde patri sablony pro vykreslovani
- napr. `app/views/users/show.html.erb`
- layouty patri do `app/views/layouts`
- railsy maji skoro na vse na co si vzpomenete helpery (`link_to`, `form_for`, ...)

db/migrate

- zde jsou migrace
- v railsech si na SQL téměř nesehneme, alespon v zakladu ne
- rake db:create
rake db:seed
rake db:migrate
rake db:rollback
rake db:drop

rails generate

- automaticke generatory kodu
- rails generate
 - scaffold
 - controller
 - model
 - migration
 - helper
 - atd. (viz. --help)

cas hrani

- pohrajeme si v terminalu

mame jeste cas?

- otazky?
- concerns
- mailers
- config/
- i18n
- deploy
- initializers
- ActiveSupport
- dalsi metody