

Instalacja Manage My Money

Robert Pankowski

Jarosław Plebański

Instalacja Manage My Money:

Robert Pankowski
Jarosław Plebański

Publication date 16 kwiecień 2009

Spis treści

1. Wstęp	1
2. Instalacja	2
2.1. Sposoby instalacji	2
2.2. Globalna instalacja aplikacji	2
Informacja	2
Prerekwizyty	2
Wymagane pakiety i gemy	2
System bazy danych	4
Źródła aplikacji	4
Konfiguracja	4
Uruchomienie migracji wszystkich baz danych	4
Uruchomienie serwera memcached	4
Przygotowywanie sphinx	5
Uruchomienie testów	5
Uruchamianie backgroundrb	5
Uruchomienie serwera aplikacji	5
Dalsze możliwości wdrożenia	5

Rozdział 1. Wstęp

Manage my money to internetowy system do zarządzania budżetem osobistym, niniejszy dokument przedstawi instrukcję krok-po-kroku instalacji systemu.

Rozdział 2. Instalacja

2.1. Sposoby instalacji

Istnieje bardzo dużo możliwych kombinacji w jakich może zostać wdrożona nasza aplikacja. Im więcej posiadasz doświadczenia z Ruby on Rails tym bardziej ten proces jest dla Ciebie oczywisty i tym więcej możliwości możesz sobie wyobrazić i pewnie dokładnie wiesz co należy zrobić aby system działał poprawnie. Tym samym niniejsza instrukcja nie jest skierowana do najbardziej doświadczonych osób, lecz raczej do początkujących, którzy chcieliby łatwo uzyskać działający system.

Zdecydowaliśmy się opisać instalację w trybie globalnym - użytkownik posiada system Ubuntu i prawa administracyjne do niego oraz pragnie zainstalować aplikację, tak by była dostępna dla wszystkich użytkowników. Jak największa część bibliotek zostanie zainstalowana globalnie, tak by mogła być w przyszłości współdzielona z innymi aplikacjami.

Ponieważ dystrybucja Ubuntu pochodzi od Debiana doświadczeni użytkownicy powinni móc bez większych problemów skorzystać z niniejszej instrukcji także w tej dystrybucji.

2.2. Globalna instalacja aplikacji

Informacja

Przy wykonywaniu wszystkich poleceń, które zawierają jako wartość parametru "production" możemy je zastąpić słowem "development", jeśli chcemy by dane polecenie zostało wykonane w trybie i na bazie deweloperskiej zamiast produkcyjnej.

Prerekwizyty

System Ubuntu 8.10, uprawnienia administratora i dostęp do internetu

Wymagane pakiety i gemy

Do działania aplikacji wymagane są następujące pakiety (lub czasem wystarczą odpowiadające im binarki przekompilowane samodzielnie):

ruby - Wymagany w wersji conajmniej 1.8.7 ale nie 1.9. Instalacja:

```
sudo apt-get install ruby-full
```

Wersja:

```
rupert@puchatek:~$ ruby -v
```

```
ruby 1.8.7 (2008-08-11 patchlevel 72) [i486-linux]
```

rubygems - Wymagane w wersji conajmniej 1.3.1 Instalacja na podstawie <https://help.ubuntu.com/community/RubyOnRails> :

```
wget http://rubyforge.org/frs/download.php/45905/rubygems-1.3.1.tgz
tar xzvf rubygems-1.3.1.tgz
cd rubygems-1.3.1
sudo ruby setup.rb
sudo ln -s /usr/bin/gem1.8 /usr/bin/gem
sudo gem update --system
cd ..
```

rails - Wymagane w wersji 2.3.2 conajmniej. Instalacja:

```
sudo gem install rails -v 2.3.2 --no-rdoc --no-ri
```

Dodatkowe wymagane gemy wraz z pakietami niezbędnymi do ich zbudowania:

```
sudo gem sources -a http://gems.github.com
```

```
sudo gem install packet --no-rdoc --no-ri
sudo gem install chronic --no-rdoc --no-ri
```

```
sudo gem install collections --no-rdoc --no-ri
```

```
sudo gem install mislav-will_paginate --no-rdoc --no-ri
```

```
sudo apt-get install libxslt-dev libxslt-ruby
sudo gem install nokogiri --no-rdoc --no-ri
```

```
sudo gem install fastercsv --no-rdoc --no-ri
```

```
# For stats only:
```

```
sudo gem install flay --no-rdoc --no-ri
sudo gem install flog --no-rdoc --no-ri
sudo gem install railroad --no-rdoc --no-ri
sudo gem install reek --no-rdoc --no-ri
sudo gem install roodi --no-rdoc --no-ri
sudo gem install rspec-rails --no-rdoc --no-ri
```

```
sudo gem install jscruggs-metric_fu --no-rdoc --no-ri
sudo gem install mergulhao-rcov --no-rdoc --no-ri
```

System cachowania danych: memcached

```
sudo apt-get install memcached
```

**Sphinx (wymagany do działania funkcjonalności podpowiadania):
Dokładna instrukcja instalacji: <http://www.sphinxsearch.com/docs/current.html#installation>**

System bazy danych

System Manage My Money został przystosowany do bazy danych PostgreSQL w wersji 8.3. Instalacja:

```
sudo apt-get install postgresql
sudo apt-get install postgresql-client
sudo apt-get install libpq-dev
sudo gem install postgres
```

Tworzenie wymaganych baz:

```
sudo su postgres
createdb money_development -E UTF8
```

```
Analogicznie dla: money_test
Analogicznie dla: money_selenium
Analogicznie dla: money_production
```

```
createuser test_3m -S -d -R -P
```

```
exit
```

Źródła aplikacji

Źródła stabilnych wersji aplikacji znajdują się pod ogólnodostępnym adresem: <http://github.com/paneq/manage-my-money/downloads> Na tej stronie należy wybrać paczkę w wybranym formacie (tgz lub zip) przy wersji 1.0. Zapisany plik rozpakować do folderu w którym chcemy zainstalować aplikację

Konfiguracja

Z kodem aplikacji dystrybuowana jest przykładowa konfiguracja, aby jej użyć zmien nazwy wszystkich plików *_example (usuając ten przyrostek) w config/ directory i ewentualnie wyedytuj ich zawartość zgodnie z potrzebami.

Uruchomienie migracji wszystkich baz danych

W konsoli (w katalogu aplikacji) wykonaj

```
rake db:migrate:all
```

Uruchomienie serwera memcached

W konsoli wykonaj

```
memcached -p NUMER_PORTU
```

Przygotowywanie sphinx

W celu wygenerowania konfiguracji dla indeksa wykonaj:

```
rake thinking_sphinx:configure RAILS_ENV=production
```

Aby utworzyć indeksy wykonaj:

```
rake thinking_sphinx:index RAILS_ENV=production
```

Aby uruchomić demona wyszukiwania wykonaj:

```
rake thinking_sphinx:start RAILS_ENV=production
```

Uruchomienie testów

Poprawne wykonanie wszystkich testów automatycznych jest warunkiem koniecznym poprawnego wdrożenia. Aby uruchomić testy w konsoli (w katalogu aplikacji) wykonaj:

```
rake test
```

Uruchamianie backgroundrb

W konsoli (w katalogu aplikacji) wykonaj:

```
./script/backgroundrb start -e production
```

Uruchomienie serwera aplikacji

W konsoli (w katalogu aplikacji) wykonaj:

```
./script/server -e production
```

Aplikacja dostępna jest pod adresem localhost:3000

Dalsze możliwości wdrożenia

Opisany został tylko najprostszy sposób uruchomienia aplikacji. Jeśli chcesz w pełni profesjonalnie zarządzać swoim wdrożeniem zapraszamy do przeczytania przynajmniej następujących materiałów:

- Capistrano to narzędzie głównie dedykowane do automatyzacji procesu wdrażania aplikacji Ruby on Rails. Z jego pomocą możesz łatwo i szybko przegrywać w odpowiednie miejsce na swoim serwerze najnowsze wersje aplikacji, restartować serwery lub przeprowadzać migracje. Szczegóły na stronie: <http://www.capify.org/getting-started/from-the-beginning/>
- Aktualnie najprostszy sposób na profesjonalne wdrożenie produkcyjne to skorzystanie z modułu mod_passenger dla serwera Apache. Jest

to proste, stabilne i zalecane rozwiązanie. Wszystkie niezbędne informacje a nawet wiele więcej możesz znaleźć w doskonałym poradniku: <http://hackd.thrivesmarthq.com/how-to-setup-a-linux-server-for-ruby-on-rails-with-github-and-phusion-passenger>

- Jeśli nie masz możliwości skorzystania z tej opcji, lub po prostu nie chcesz możesz też wykorzystać jako mongrel jako serwer aplikacji. Zajrzyj na: <http://mongrel.rubyforge.org/>
- Jeśli udało Ci się zdobyć tylu użytkowników, że jeden mongrel to za mało skorzystaj z całego klastra: <http://mongrel.rubyforge.org/wiki/MongrelCluster>
- Jeśli potrzebujesz jeszcze więcej to znaczy, że ta dokumentacja nie jest przeznaczona dla Ciebie. Z pewnością bardziej zaawansowane materiały nt. Ruby on Rails i jego skalowania oraz optymalizacji znajdziesz w Internecie. Z naszej strony możemy polecić: <http://railslab.newrelic.com/scaling-rails>