

A Rails keretrendszer Gyakorlat

Kovács Gábor

2011. október 4.

A Rails keretrendszert egy virtuális környezetben ¹ futó Debian Linuxra ² telepítjük a gyakorlat folyamán. Ez a dokumentáció kiindulási pontként egy ilyen, grafikus felhasználói környezetet is tartalmazó előtelepített operációs rendszert feltételez.

A Rails környezetünk back-endje hosszú távon egy MySQL adatbázis kezelő lesz. Ezt az `apt-get install mysql-server` parancs konzolon történő kiadásával telepíthetjük, a telepítő többször rá fog kérdezni a root felhasználó jelszavára, azt tetszés szerinti értékre állíthatjuk, azonban jegyezzük meg, mert a Rails adatbázis konfigurációs fájljában ezt meg kell adnunk. A gyakorlatra telepített környezetben az egyszerűség kedvéért üres jelszót fogunk használni.

A Rails adatbázis adapterei fordításához szükségünk van C és C++ fordítóra, valamint a fejlesztői függvénykönyvtárakra, amelyeket az `apt-get install gcc` és `apt-get install g++` parancsok konzolon való kiadásával telepíthetünk.

A Rails környezet telepítéséhez egy Ruby környezet szükséges, ami mellé kiegészítőként telepítjük az RI dokumentációs rendszert és a natív kiegészítések fordításához szükséges Ruby fejlesztői függvénykönyvtárakat. Ezt a következő paranccsal telepíthetjük rendszergazdaként:

```
kovacs@debian:~$ sudo bash
[sudo] password for kovacs:
root@debian:/home/kovacs# apt-get install ruby1.9.1 ri1.9.1 ruby1.9.1-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer
required:
```

¹Oracle VirtualBox (<http://www.virtualbox.org>) az otthonra javasolt virtualizációs eszköz, viszont használható a vmware, illetve a parallels is

²Mivel az Ubuntu Linux csomagkezelője megegyezik a Debian Linux disztribúció csomagkezelőjével, a dokumentumban felsorolt csomagnevek egy az egyben átvehetők. Más disztribúciók esetén a csomagok nevei különbözhetnek.

```

linux-kbuild-2.6.32 linux-headers-2.6.32-5-common gcc-4.3-base cpp-4.3
Use 'apt-get autoremove' to remove them.
The following extra packages will be installed:
  libruby1.9.1 libyaml-0-2
Suggested packages:
  ruby1.9.1-examples graphviz
The following NEW packages will be installed:
  libruby1.9.1 libyaml-0-2 ri1.9.1 ruby1.9.1 ruby1.9.1-dev
0 upgraded, 5 newly installed, 0 to remove and 746 not upgraded.
Need to get 0 B/9,405 kB of archives.
After this operation, 64.9 MB of additional disk space will be used.
Do you want to continue [Y/n]?
Selecting previously deselected package libyaml-0-2.
(Reading database ... 121123 files and directories currently installed.)
Unpacking libyaml-0-2 (from .../libyaml-0-2_0.1.4-1_amd64.deb) ...
Selecting previously deselected package libruby1.9.1.
Unpacking libruby1.9.1 (from .../libruby1.9.1_1.9.2.290-2_amd64.deb) ...
Selecting previously deselected package ruby1.9.1.
Unpacking ruby1.9.1 (from .../ruby1.9.1_1.9.2.290-2_amd64.deb) ...
Selecting previously deselected package ri1.9.1.
Unpacking ri1.9.1 (from .../ri1.9.1_1.9.2.290-2_all.deb) ...
Selecting previously deselected package ruby1.9.1-dev.
Unpacking ruby1.9.1-dev (from .../ruby1.9.1-dev_1.9.2.290-2_amd64.deb) ...
Processing triggers for man-db ...
Setting up libyaml-0-2 (0.1.4-1) ...
Setting up libruby1.9.1 (1.9.2.290-2) ...
Setting up ruby1.9.1 (1.9.2.290-2) ...
update-alternatives: using /usr/bin/gem1.9.1 to provide /usr/bin/gem (gem)
  in auto mode.
update-alternatives: using /usr/bin/ruby1.9.1 to provide /usr/bin/ruby (ruby
  ) in auto mode.
Setting up ri1.9.1 (1.9.2.290-2) ...
Setting up ruby1.9.1-dev (1.9.2.290-2) ...

```

Ezután ellenőrizzük, hogy elérhető-e a ruby értelmező és a gem Ruby csomagkezelő.

```

root@debian:/home/kovacsg# ruby --version
ruby 1.9.2p290 (2011-07-09 revision 32553) [x86_64-linux]
root@debian:/home/kovacsg# gem --version
1.3.7

```

A következő lépés a Rails keretrendszer telepítése, amit a gem Ruby csomagkezelővel teszünk meg. Ez egyben feltelepíti a legfontosabb kiegészítőket is, illetve a hozzájuk kapcsolódó dokumentációt.

```

root@debian:/home/kovacsg# gem install rails
root@debian:/home/kovacsg# gem install rails
Building native extensions. This could take a while...
Depending on your version of ruby, you may need to install ruby rdoc/ri data
:

<= 1.8.6 : unsupported
= 1.8.7 : gem install rdoc-data; rdoc-data --install
= 1.9.1 : gem install rdoc-data; rdoc-data --install
>= 1.9.2 : nothing to do! Yay!
Successfully installed multi_json-1.0.3
Successfully installed activesupport-3.1.0
Successfully installed builder-3.0.0
Successfully installed i18n-0.6.0

```

```

Successfully installed bcrypt-ruby-3.0.1
Successfully installed activemodel-3.1.0
Successfully installed rack-1.3.4
Successfully installed rack-cache-1.0.3
Successfully installed rack-test-0.6.1
Successfully installed rack-mount-0.8.3
Successfully installed hike-1.2.1
Successfully installed tilt-1.3.3
Successfully installed sprockets-2.0.2
Successfully installed erubis-2.7.0
Successfully installed actionpack-3.1.0
Successfully installed arel-2.2.1
Successfully installed tzinfo-0.3.30
Successfully installed activerecord-3.1.0
Successfully installed activeresource-3.1.0
Successfully installed mime-types-1.16
Successfully installed polyglot-0.3.2
Successfully installed treetop-1.4.10
Successfully installed mail-2.3.0
Successfully installed actionmailer-3.1.0
Successfully installed rake-0.9.2
Successfully installed thor-0.14.6
Successfully installed rack-ssl-1.3.2
Successfully installed rdoc-3.9.4
Successfully installed railties-3.1.0
Successfully installed bundler-1.0.21
Successfully installed rails-3.1.0
31 gems installed
Installing ri documentation for multi_json-1.0.3...
Installing ri documentation for activesupport-3.1.0...
Installing ri documentation for builder-3.0.0...
unable to convert "\xF1" from ASCII-8BIT to UTF-8 for README, skipping
unable to convert "\xF1" from ASCII-8BIT to UTF-8 for README.rdoc, skipping
Installing ri documentation for i18n-0.6.0...
Installing ri documentation for bcrypt-ruby-3.0.1...
Installing ri documentation for activemodel-3.1.0...
Installing ri documentation for rack-1.3.4...
Installing ri documentation for rack-cache-1.0.3...
Installing ri documentation for rack-test-0.6.1...
Installing ri documentation for rack-mount-0.8.3...
Installing ri documentation for hike-1.2.1...
Installing ri documentation for tilt-1.3.3...
Installing ri documentation for sprockets-2.0.2...
Installing ri documentation for erubis-2.7.0...
Installing ri documentation for actionpack-3.1.0...
Installing ri documentation for arel-2.2.1...
Installing ri documentation for tzinfo-0.3.30...
Installing ri documentation for activerecord-3.1.0...
Installing ri documentation for activeresource-3.1.0...
Installing ri documentation for mime-types-1.16...
Installing ri documentation for polyglot-0.3.2...
Installing ri documentation for treetop-1.4.10...
Installing ri documentation for mail-2.3.0...
Installing ri documentation for actionmailer-3.1.0...
Installing ri documentation for rake-0.9.2...
Installing ri documentation for thor-0.14.6...
Installing ri documentation for rack-ssl-1.3.2...
Installing ri documentation for rdoc-3.9.4...
Installing ri documentation for railties-3.1.0...
Installing ri documentation for bundler-1.0.21...
Installing ri documentation for rails-3.1.0...
file 'lib' not found

```

```

Installing RDoc documentation for multi_json-1.0.3...
Installing RDoc documentation for activesupport-3.1.0...
Installing RDoc documentation for builder-3.0.0...
unable to convert "\xF1" from ASCII-8BIT to UTF-8 for README, skipping
unable to convert "\xF1" from ASCII-8BIT to UTF-8 for README.rdoc, skipping
Installing RDoc documentation for i18n-0.6.0...
Installing RDoc documentation for bcrypt-ruby-3.0.1...
Installing RDoc documentation for activemodel-3.1.0...
Installing RDoc documentation for rack-1.3.4...
Installing RDoc documentation for rack-cache-1.0.3...
Installing RDoc documentation for rack-test-0.6.1...
Installing RDoc documentation for rack-mount-0.8.3...
Installing RDoc documentation for hike-1.2.1...
Installing RDoc documentation for tilt-1.3.3...
Installing RDoc documentation for sprockets-2.0.2...
Installing RDoc documentation for erubis-2.7.0...
Installing RDoc documentation for actionpack-3.1.0...
Installing RDoc documentation for arel-2.2.1...
Installing RDoc documentation for tzinfo-0.3.30...
Installing RDoc documentation for activerecord-3.1.0...
Installing RDoc documentation for activerecord-3.1.0...
Installing RDoc documentation for mime-types-1.16...
Installing RDoc documentation for polyglot-0.3.2...
Installing RDoc documentation for treetop-1.4.10...
Installing RDoc documentation for mail-2.3.0...
Installing RDoc documentation for actionmailer-3.1.0...
Installing RDoc documentation for rake-0.9.2...
Installing RDoc documentation for thor-0.14.6...
Installing RDoc documentation for rack-ssl-1.3.2...
Installing RDoc documentation for rdoc-3.9.4...
Installing RDoc documentation for railties-3.1.0...
Installing RDoc documentation for bundler-1.0.21...
Installing RDoc documentation for rails-3.1.0...
file 'lib' not found

```

A következő lépésben az adatbáziskezelő Ruby illesztőjét telepítjük. Az adatbázis adapterek natív kiterjesztései fordításához két további fejlesztői függvénykönyvtárat kell telepítenünk az operációs rendszer csomagkezelőjével, a libsqlite3-dev és a libmysqlclient-dev nevű csomagot ³:

```

root@debian:/home/kovacs# apt-get install libsqlite3-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer
required:
  linux-kbuild-2.6.32 linux-headers-2.6.32-5-common gcc-4.3-base cpp-4.3
Use 'apt-get autoremove' to remove them.
Suggested packages:
  sqlite3-doc
The following NEW packages will be installed:
  libsqlite3-dev
0 upgraded, 1 newly installed, 0 to remove and 746 not upgraded.
Need to get 0 B/536 kB of archives.
After this operation, 1,315 kB of additional disk space will be used.
Selecting previously deselected package libsqlite3-dev.
(Reading database ... 132741 files and directories currently installed.)

```

³Más Linux disztribúciókon ezek neve eltérhet. Windows-on elég az előre lefordított DLL-t elhelyezni az elérési úton

```

Unpacking libsqlite3-dev (from .../libsqlite3-dev_3.7.8-1_amd64.deb) ...
Setting up libsqlite3-dev (3.7.8-1) ...
root@debian:/home/kovacsg# apt-get install libmysqlclient-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer
required:
  linux-kbuild-2.6.32 linux-headers-2.6.32-5-common gcc-4.3-base cpp-4.3
Use 'apt-get autoremove' to remove them.
The following extra packages will be installed:
  zlib1g-dev
The following NEW packages will be installed:
  libmysqlclient-dev zlib1g-dev
0 upgraded, 2 newly installed, 0 to remove and 746 not upgraded.
Need to get 0 B/3,447 kB of archives.
After this operation, 10.3 MB of additional disk space will be used.
Do you want to continue [Y/n]?
Selecting previously deselected package zlib1g-dev.
(Reading database ... 132752 files and directories currently installed.)
Unpacking zlib1g-dev (from .../zlib1g-dev_1%3a1.2.3.4.dfsg-3_amd64.deb) ...
Selecting previously deselected package libmysqlclient-dev.
Unpacking libmysqlclient-dev (from .../libmysqlclient-dev_5.1.58-1_amd64.deb
) ...
Processing triggers for man-db ...
Setting up zlib1g-dev (1:1.2.3.4.dfsg-3) ...
Setting up libmysqlclient-dev (5.1.58-1) ...

```

Ezután telepíthetjük először az SQLite, majd a MySQL adapterének Ruby API-ját:

```

root@debian:/home/kovacsg# gem install sqlite3
Building native extensions. This could take a while...
Successfully installed sqlite3-1.3.4
1 gem installed
Installing ri documentation for sqlite3-1.3.4...
Installing RDoc documentation for sqlite3-1.3.4...
root@debian:/home/kovacsg# gem install mysql2
Building native extensions. This could take a while...
Successfully installed mysql2-0.3.7
1 gem installed
Installing ri documentation for mysql2-0.3.7...
Enclosing class/module 'mMysql2' for class Client not known
Enclosing class/module 'mMysql2' for class Result not known
Enclosing class/module "cMysql2Result" for alias size count not known
Installing RDoc documentation for mysql2-0.3.7...
Enclosing class/module 'mMysql2' for class Client not known
Enclosing class/module 'mMysql2' for class Result not known
Enclosing class/module "cMysql2Result" for alias size count not known

```

A Rails 3.1-es verziója változást hozott a JavaScript kezelésben, amely két további gem telepítését teszi szükségessé. Az `execjs` JavaScriptek Rubyba ágyzását teszi lehetővé, a `therubyracer` pedig egy JavaScript interpreter.

```

root@debian:/home/kovacsg# gem install execjs
Successfully installed execjs-1.2.9
1 gem installed
Installing ri documentation for execjs-1.2.9...
Installing RDoc documentation for execjs-1.2.9...
root@debian:/home/kovacsg# gem install therubyracer

```

```

Building native extensions. This could take a while...
Successfully installed libv8-3.3.10.2-x86_64-linux
Successfully installed therubyracer-0.9.4
2 gems installed
Installing ri documentation for libv8-3.3.10.2-x86_64-linux...
Installing ri documentation for therubyracer-0.9.4...
Installing RDoc documentation for libv8-3.3.10.2-x86_64-linux...
Installing RDoc documentation for therubyracer-0.9.4...

```

A Rails környezetünket létrehoztuk, a következő lépés annak ellenőrzése, hogy a rails parancs elérhető-e. Ha nem, akkor adjuk hozzá a PATH környezeti változóhoz, majd ismétljük meg az ellenőrzést.

```

kovacs@debian:~$ rails -v
bash: rails: command not found
kovacs@debian:~$ export PATH=$PATH:/var/lib/gems/1.9.1/bin
kovacs@debian:~$ rails -v
Rails 3.0.5

```

A Rails a 3.1-es verziótól telepítés után automatikusan rákerül az elérési útra.

```

root@debian:/home/kovacs# which rails
/usr/local/bin/rails
root@debian:/home/kovacs# rails -v
Rails 3.1.0

```

A -d kapcsoló különös jelentőséggel bír számunkra, ezzel adhatjuk meg a használni kívánt adatbáziskezelő típusát, ami lehet mysql, oracle, postgresql, sqlite3, frontbase vagy ibm_db. A gyakorlatok keretében ezek közül a sqlite-ot mint alapértelmezett adatbáziskezelőt és a mysql-t fogjuk használni.

```

kovacs@debian:~$ rails
Usage:
  rails new APP_PATH [options]

Options:
  -r, [--ruby=PATH]                # Path to the Ruby binary of your choice
                                   # Default: /usr/bin/ruby1.9.1
  -b, [--builder=BUILDER]          # Path to a application builder (can be a
                                   filesystem path or URL)
  -m, [--template=TEMPLATE]       # Path to an application template (can be a
                                   filesystem path or URL)
  --skip-gemfile                   # Don't create a Gemfile
  --skip-bundle                   # Don't run bundle install
  -G, [--skip-git]                 # Skip Git ignores and keeps
  -O, [--skip-active-record]       # Skip Active Record files
  -S, [--skip-sprockets]          # Skip Sprockets files
  -d, [--database=DATABASE]       # Preconfigure for selected database (
                                   options: mysql/oracle/postgresql/sqlite3/frontbase/ibm_db/jdbcmysql/
                                   jdbcsqlite3/jdbcpostgresql/jdbc)
                                   # Default: sqlite3
  -j, [--javascript=JAVASCRIPT]   # Preconfigure for selected JavaScript
                                   library
                                   # Default: jquery
  -J, [--skip-javascript]         # Skip JavaScript files
  --dev                             # Setup the application with Gemfile
                                   pointing to your Rails checkout
  --edge                             # Setup the application with Gemfile
                                   pointing to Rails repository
  -T, [--skip-test-unit]          # Skip Test::Unit files
  --old-style-hash                 # Force using old style hash (:foo => 'bar'
                                   ') on Ruby >= 1.9

Runtime options:

```

```
-f, [--force] # Overwrite files that already exist
-p, [--pretend] # Run but do not make any changes
-q, [--quiet] # Suppress status output
-s, [--skip] # Skip files that already exist
```

Rails options:

```
-h, [--help] # Show this help message and quit
-v, [--version] # Show Rails version number and quit
```

Description:

The 'rails new' **command** creates a new Rails application with a default directory structure and configuration at the path you specify.

Example:

```
rails new ~/Code/Ruby/weblog
```

This generates a skeletal Rails installation in ~/Code/Ruby/weblog. See the README in the newly created application to get going.

A Rails keretrendszerben a rails parancs az az univerzális eszköz mellyel többek között új komponenseket hozhatunk létre és elindíthatjuk a beépített webservert. Egy új Rails alkalmazást a rails parancsnak new opciót megadva hozhatunk létre, a második argumentum az alkalmazás neve. A parancsot kiadva a konzolon láthatjuk az automatikusan generált fájlokat. Az egyes könyvtárak értelmezéséről előadáson esett szó, e gyakorlat keretében megnézzük az egyes fájlok szerepét és tartalmát.

```
kovacs@debian:~$ rails new gyakorlat
create
create  README
create  Rakefile
create  config.ru
create  .gitignore
create  Gemfile
create  app
create  app/assets/images/rails.png
create  app/assets/javascripts/application.js
create  app/assets/stylesheets/application.css
create  app/controllers/application_controller.rb
create  app/helpers/application_helper.rb
create  app/mailers
create  app/models
create  app/views/layouts/application.html.erb
create  app/mailers/.gitkeep
create  app/models/.gitkeep
create  config
create  config/routes.rb
create  config/application.rb
create  config/environment.rb
create  config/environments
create  config/environments/development.rb
create  config/environments/production.rb
create  config/environments/test.rb
create  config/initializers
create  config/initializers/backtrace_silencers.rb
create  config/initializers/inflections.rb
create  config/initializers/mime_types.rb
create  config/initializers/secret_token.rb
create  config/initializers/session_store.rb
create  config/initializers/wrap_parameters.rb
create  config/locales
create  config/locales/en.yml
create  config/boot.rb
create  config/database.yml
```

```

create db
create db/seeds.rb
create doc
create doc/README_FOR_APP
create lib
create lib/tasks
create lib/tasks/.gitkeep
create lib/assets
create lib/assets/.gitkeep
create log
create log/.gitkeep
create public
create public/404.html
create public/422.html
create public/500.html
create public/favicon.ico
create public/index.html
create public/robots.txt
create script
create script/rails
create test/fixtures
create test/fixtures/.gitkeep
create test/functional
create test/functional/.gitkeep
create test/integration
create test/integration/.gitkeep
create test/unit
create test/unit/.gitkeep
create test/performance/browsing_test.rb
create test/test_helper.rb
create tmp/cache
create tmp/cache/assets
create vendor/assets/stylesheets
create vendor/assets/stylesheets/.gitkeep
create vendor/plugins
create vendor/plugins/.gitkeep

```

Az `app` könyvtár fogja tartalmazni az általunk készített Ruby és beágyazott Ruby kódot tartalmazó HTML forrásokat, amelyeket a MVC minta alapján struktúrál a Rails, amelyeket a `test` könyvtárban elhelyezett teszt osztályokkal ellenőrizzük. A `config` könyvtár a Rails alkalmazásunk konfigurációs beállításait tartalmazza. A `db` könyvtár az aktuális adatbázis sémát, az összes eddig adatbázis séma migrációt és `sqlite` adatbáziskezelő esetén szerializált formában magát az adatbázis tartalmazza. A `doc` a dokumentációk gyűjtőhelye, a `lib` és `vendor` könyvtárak mások által készített Ruby, illetve Rails függvénykönyvtárakat tartalmazhatnak. A `public` könyvtár a beépített webservert területe, az összes ott bekövetkezett esemény a `log` könyvtárban található az aktuális Rails környezetnek megfelelő állományban kerülnek naplózásra, a webservert `tmp` könyvtárban helyezheti el az átmeneti fájlokat, mint például `session` azonosítókat, `sütiket`. A `script` könyvtár elérhetővé teszi számunkra a `rails` parancsot. A `Gemfile` és `Rakefile` a Rails alkalmazásunk által használt Ruby függvénykönyvtárakat specifikálja, amelyek vagy a telepített Ruby és Rails környezeten vagy a már említett `lib` és `vendor` könyvtárakban érhetők el.

A létrehozott `gyakorlat` nevű Rails alkalmazás könyvtárába belépve a konzolon kiadott `bundle install` paranccsal telepíthetjük az összes függőséget, az esetlegesen hiányzó vagy nem megfelelő verziójú Ruby függvénykönyvtárat. Rails 3.1-es verziótól kezdve a `new` hatására automatikusan lefut.

```

run bundle install
Fetching source index for http://rubygems.org/
Could not reach rubygems repository http://rubygems.org/
Using rake (0.9.2)
Using multi_json (1.0.3)
Using activesupport (3.1.0)
Using bcrypt-ruby (3.0.1)
Using builder (3.0.0)
Using i18n (0.6.0)

```



```

Using activemodel (3.1.0)
Using erubis (2.7.0)
Using rack (1.3.4)
Using rack-cache (1.0.3)
Using rack-mount (0.8.3)
Using rack-test (0.6.1)
Using hike (1.2.1)
Using tilt (1.3.3)
Using sprockets (2.0.2)
Using actionpack (3.1.0)
Using mime-types (1.16)
Using polyglot (0.3.2)
Using treetop (1.4.10)
Using mail (2.3.0)
Using actionmailer (3.1.0)
Using arel (2.2.1)
Using tzinfo (0.3.30)
Using activerecord (3.1.0)
Using activeresource (3.1.0)
Using activeresource (3.1.0)
Using ansi (1.3.0)
Using bundler (1.0.21)
Using coffee-script-source (1.1.2)
Using execjs (1.2.9)
Using coffee-script (2.2.0)
Using rack-ssl (1.3.2)
Using rdoc (3.9.4)
Using thor (0.14.6)
Using railties (3.1.0)
Using coffee-rails (3.1.1)
Using jquery-rails (1.0.14)
Using rails (3.1.0)
Using sass (3.1.8)
Using sass-rails (3.1.4)
Using sqlite3 (1.3.4)
Using turn (0.8.2)
Using uglifier (1.0.3)
Your bundle is complete! Use 'bundle show [gemname]' to see where a bundled
gem is installed.

```

A 3.1-es verzió JavaScript változásai miatt módosítanunk kell a Gemfile állományt megadva a therubyracer-t mint interpretert. Az állomány tetszőleges helyére a következő sort szúrjuk be:

```
gem 'therubyracer', :platforms => :ruby
```

A beépített webservert a rails server opciójával indíthatjuk el a Rail alkalmazásunk gyökérkönyvtárából. Ha a szervert háttéralkalmazásként kívánjuk futtatni, akkor a -d kapcsolót is hozzá kell fűznünk a parancssorhoz. Ezután a szerver elérhető a http://localhost:3000 webcímen, lásd 1. ábra.

```

kovacs@debian:~/gyakorlat/app/views/say$ rails server
=> Booting WEBrick
=> Rails 3.1.0 application starting in development on http://0.0.0.0:3000
=> Call with -d to detach
=> Ctrl-C to shutdown server
[2011-10-04 13:00:16] INFO WEBrick 1.3.1
[2011-10-04 13:00:16] INFO ruby 1.9.2 (2011-07-09) [x86_64-linux]
[2011-10-04 13:00:16] INFO WEBrick::HTTPServer#start: pid=8867 port=3000

```

Két fontos konfigurációs fájl érdemel most említést: a adatbázis hozzáférés konfigurációjára szolgáló database.yml és az URL leképezéseket definiáló routes.rb.

Mivel a Rails alkalmazás létrehozásakor nem adtunk meg, hogy melyik adatbáziskezelőt kívánjuk használni, ezért az alapértelmezett SQLite konfigurációs paraméterei jelennek meg a database.yml-ben mindhárom definiált környezethez, vagyis a fejlesztési (development), tesztelési (test) és éles (production).

Az egyes környezetek adapter opciója határozza meg az adatbáziskezelő típusát és a használható paraméterkészletet. Az SQLite az adatait fájlokban tárolja, amelyek a Rails alkalmazás db könyvtárában találhatóak, és a környezet nevével azonosítottak e fájl szerint.

```
# SQLite version 3.x
# gem install sqlite3
#
# Ensure the SQLite 3 gem is defined in your Gemfile
# gem 'sqlite3'
development:
  adapter: sqlite3
  database: db/development.sqlite3
  pool: 5
  timeout: 5000

# 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:
  adapter: sqlite3
  database: db/test.sqlite3
  pool: 5
  timeout: 5000

production:
  adapter: sqlite3
  database: db/production.sqlite3
  pool: 5
  timeout: 5000
```

MySQL esetén a mysql2 adaptert használjuk. Fontos opció az alapértelmezett karakterkódolás megadása, ami, ha lehet hagyjunk változatlanul utf8 értéken. A `database` opció a séma nevét tartalmazza értékként. A felhasználónév és jelszó megadása kötelező. A kapcsolat lehet `socket` vagy `host` és `port` alapon megadott.

```
# MySQL. Versions 4.1 and 5.0 are recommended.
#
# 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:
# http://dev.mysql.com/doc/refman/5.0/en/old-client.html
development:
  adapter: mysql2
  encoding: utf8
  reconnect: false
  database: gyakorlat_development
  pool: 5
  username: root
  password:
  socket: /var/run/mysqld/mysqld.sock

# 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:
  adapter: mysql2
  encoding: utf8
  reconnect: false
```

```

database: gyakorlat_test
pool: 5
username: root
password:
socket: /var/run/mysqld/mysqld.sock

production:
adapter: mysql2
encoding: utf8
reconnect: false
database: gyakorlat_production
pool: 5
username: root
password:
socket: /var/run/mysqld/mysqld.sock

```

Az adatbáziskapcsolatnak a szerver elindítása előtt léteznie kell. Ez SQLite esetén nem probléma, MySQL esetén viszont a sémákat explicite létre kell hoznunk. Ez a következő paranccsal történhet meg, amely a development és a test környezetekhez hozza létre a sémát.

```
kovacs@debian:~/gyakorlat$ rake db:create
```

A másik leírónk azt adja meg, hogy milyen struktúrájú legyen az URL, amivel elérjük a Rails alkalmazásunk egyes funkcióit. Az alábbi kódrészlet a legáltalánosabb beállítást tartalmazza. A webservert IP címe után a Controller osztály neve (`:controller`), majd a Controller osztály egy metódusa (`:action`), majd egy adatbázis azonosító (`:id`), és végül formázási útmutató következik, például `.html` vagy `.xml`. Az utolsó három megadása opcionális.

```

Feladat::Application.routes.draw do
  match ':controller(/:action(/:id(.:format)))'
end

```

Nézzük meg, hogy miként tudunk dinamikus tartalmat létrehozni Rails-szel. A példák a [1] könyvből valók.

Hozunk létre egy új controllert a `rails generate` opciójával. A második argumentum (`controller`) azt mondja meg, hogy egy új controllert hozunk létre, a harmadik a controller nevét. A negyedik és minden további paraméter a controllerben definiál akciót. E parancs négy Ruby forrásfájlt és egy könyvárat hoz létre az akcióknak megfelelő weboldalak, view-k számára. A controller nevének megfelelő controller osztályt (`say_controller.rb`), helper osztályt, illetve ezek funkcionális és egységtesztjéhez használható osztályokat.

```

ovacs@debian:~/gyakorlat$ rails generate controller say
create  app/controllers/say_controller.rb
invoke  erb
create  app/views/say
invoke  test_unit
create  test/functional/say_controller_test.rb
invoke  helper
create  app/helpers/say_helper.rb
invoke  test_unit
create  test/unit/helpers/say_helper_test.rb
invoke  assets
invoke  coffee
create  app/assets/javascripts/say.js.coffee
invoke  scss
create  app/assets/stylesheets/say.css.scss

```

Bármilyen tartalom megjelenítéséhez a `routes.rb` alapján a view könyvtárban kell elhelyeznünk az akciónak megfelelő néven egy beágyazott Ruby kódot tartalmazó HTML fájlt (`.rhtml` vagy `.html.erb`).

Az alkalmazás keretét az `app/views/layouts/application.html.erb` fájl definiálja, amely a HTML dokumentum törzs helyén egy beágyazott `yield` parancsot tartalmaz, amely átadja a vezérlést az akció HTML-ének, ami jelen esetben `hello` lesz.

Ez alapján nézzük meg a szokásos Hello, world alkalmazást ezúttal Rails-ben. Az `app/views/say` könyvtárban létrehozunk egy `hello.html.erb` nevű view-t, amely a `say` controller `hello` akciójához kötődik. Az eredményt a `http://localhost:3000/say/hello` linken ellenőrizhetjük.

```
<h1>Hello , world!</h1>
```

Ez dinamikussá tehetjük az aktuális idő kírásával.

```
<h1>Hello , world!</h1>
<%= Time.now %>
```

Mivel a nézetbe nem illik logikát rakni, csak a megjelenítendő értéket, ezért áttesszük az idő lekérdezését a kontrollerbe, annak is az akciónak megfelelő metódusába, a `hello`-ba

```
class SayController < ApplicationController
  def hello
    @time=Time.now
  end
end
```

A nézetben pedig csak hivatkozunk a Controller példányváltozóra.

```
<h1>Hello , world!</h1>
<%= @time %>
```

Az oldalak akciói átadhatják egymásnak a vezérlést, ehhez linkeket definálunk a `link_to` metódussal. Az alábbi példában a link neve "Hello, world" lesz, a kapcsolódó akció pedig `helloworld` névre hallgat. Ez utóbbihoz definiálnunk kell egy új metódust a Controller osztályban, és egy a View-t megvalósító HTML oldalt.

```
<h1><%= link_to "Hello ,_world", :action => "helloworld" %></h1>
<%= @time %>
```

A controllernek és akciónak megfelelő megjelenített oldalt a 2. ábra mutatja. Az oldal forrását megnézve felismerjük benne a layout által nyújtott keretet és a View beágyazott kódját.

```
<!DOCTYPE html>
<html>
<head>
  <title>Gyakorlat</title>
  <link href="/assets/application.css?body=1" media="screen" rel="stylesheet"
    type="text/css" />
  <link href="/assets/say.css?body=1" media="screen" rel="stylesheet" type="
    text/css" />
  <script src="/assets/jquery.js?body=1" type="text/javascript"></script>
  <script src="/assets/jquery_ujs.js?body=1" type="text/javascript"></script>
  <script src="/assets/say.js?body=1" type="text/javascript"></script>
  <script src="/assets/application.js?body=1" type="text/javascript"></script>
  <meta content="authenticity_token" name="csrf-param" />
  <meta content="Yo+SJluJxsf1V7AaNzJQCrvMhDACed7a9tv7q8HZeVQ=" name="csrf-
    token" />
</head>
<body>

  <h1>Hello , world!</h1>
  2011-10-04 13:04:53 +0200

</body>
</html>
```

A Rails MVC filozófiájának harmadik eleme a modell, amelyet szintén a `rails` parancs `generate` opciójával hozhatunk létre. A harmadik argumentum a modell osztály neve, amely a konvenció alapján egy egyes számban megadó és a szavakat `_` szimbólummal összefűző string. Ennek többes számú változatával

jön létre az az adatbázisban egy tábla. A parancs kiadása négy fájlt hoz létre: egy adatbázis migrációs Ruby szkriptet, egy ActiveRecord::Base leszármazottat a modell osztályok közé, egy egységtesztet tartalmazó osztályt és egy tesztadatokat tartalmazó YAML fájlt.

```
kovacs@debian:~/gyakorlat/app/views/say$ rails generate model user
invoke active_record
create db/migrate/20111004111050_create_users.rb
create app/models/user.rb
invoke test_unit
create test/unit/user_test.rb
create test/fixtures/users.yml
```

Az adatbázis migrációs szkriptben az adatmodell változtatásait adjuk meg. A modell létrehozása után egy olyan táblát hozna létre az alapértelmezett szkript, amelyben egy azonosító és két időpecsét oszlop szerepelne. A feladatunk az, hogy ezt kiegészítsük az általunk tárolni kívánt adatokkal. A tábla neve users, ehhez hozzáadunk három string típusú rendre a username, password és email azonosítókkal. A változás most egy a create_table metódus által létrehozandó tábla hat attribútummal.

```
def change
  create_table :users do |t|
    t.string :username
    t.string :password
    t.string :email
    t.timestamps
  end
end
```

Ha a Rails alkalmazás létrehozásak megadtuk volna a mysql adatbázis opciót, akkor először létre kell hoznunk a megfelelő táblát, majd utána elvégezhethetjük a tábla struktúrájának módosítását. Ezekben a rake parancs nyújt segítséget. A rake db:create létrehozza a táblákat, ha még nem tettül volna meg, a rake db:migrate módosítja a séma struktúráját.

```
kovacs@debian:~/gyakorlat_mysql/db/migrate$ rake db:create
(in /home/kovacs/gyakorlat_mysql)
kovacs@debian:~/gyakorlat/db/migrate$ rake db:migrate
(in /home/kovacs/gyakorlat)
== CreateUsers: migrating
-----
-- create_table(:users)
--> 0.0027s
== CreateUsers: migrated (0.0030s)
-----
```

A gyakorlaton a mysql illesztővel létrehozott Rails alkalmazás a migráció után a 3. ábrán látható táblát hozta létre.

Az adatokon végzett manipulációt a rails console paranccsal előhívható interaktív Ruby konzolon (irb) illusztráljuk. A User osztály konstruktorának argumentumai között a tábla attribútumaira a migrációs szkriptben megadott szimbólumokkal hivatkozhatunk, és rendelhetünk hozzá értéket. A nem inicializált attribútumok null értéket vesznek fel.

```
kovacs@debian:~/gyakorlat_mysql/db/migrate$ rails console
kovacs@debian:~/gyakorlat/db/migrate$ rails console
Loading development environment (Rails 3.1.0)
irb(main):001:0> u = User.new
=> #<User id: nil, username: nil, password: nil, email: nil, created_at: nil, updated_at: nil>
irb(main):002:0> u.password='titok'
=> "titok"
irb(main):003:0> u.username='Senki'
=> "Senki"
irb(main):004:0> u.email='senki@mail.bme.hu'
=> "senki@mail.bme.hu"
irb(main):005:0> u
```

```

=> #<User id: nil, username: "Senki", password: "titok", email: "senki@mail.
    bme.hu", created_at: nil, updated_at: nil>
irb(main):006:0> u.save
(0.1ms) BEGIN
SQL (0.5ms) INSERT INTO 'users' ('created_at', 'email', 'password', '
    updated_at', 'username') VALUES ('2011-10-04 11:20:00', 'senki@mail.
    bme.hu', 'titok', '2011-10-04 11:20:00', 'Senki')
(4.6ms) COMMIT
=> true

```

A `save` metódussal elmentettük a létrehozott példányt az adatbázisba, amelyet az adatbáziskezelőn ellenőrizhetünk, amit a 4. ábra is mutat.

Végül meggyőződhetünk arról, hogy az adatbázisban módosított érték (a `deadline` attribútum értéke) a modellben is módosul.

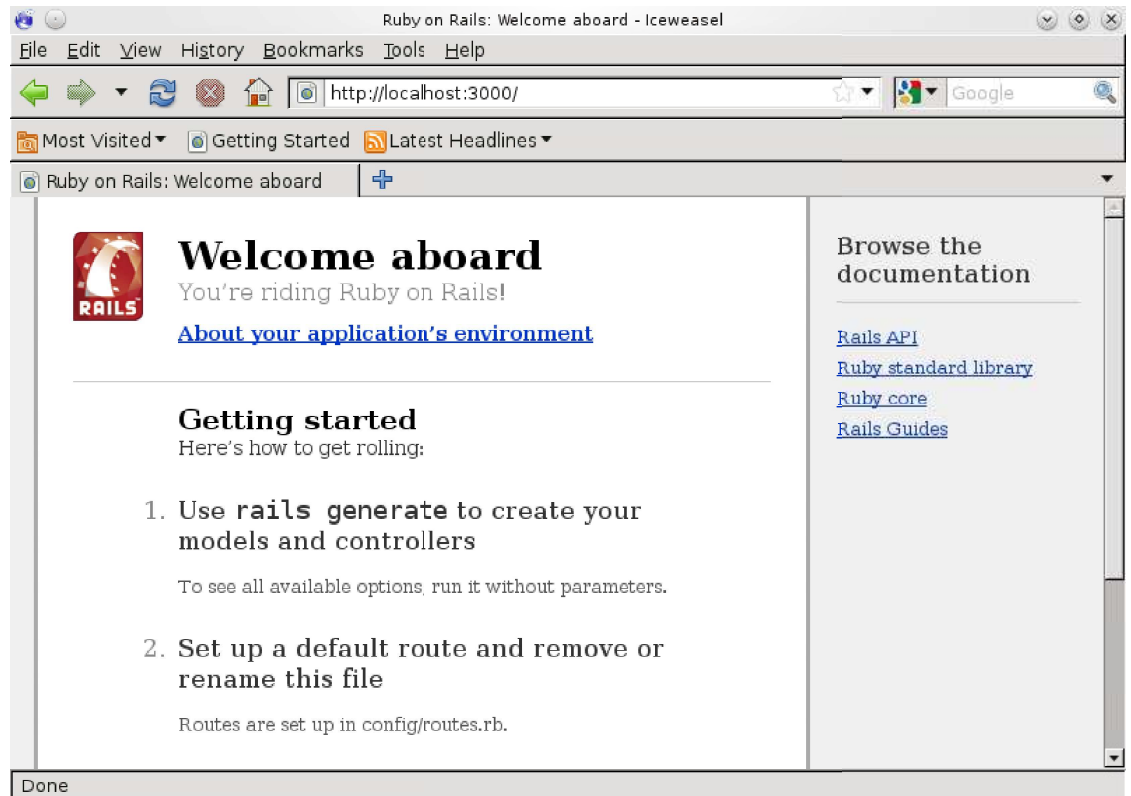
```

kovacs@debian:~/gyakorlat_mysql/db/migrate$ rails console
Loading development environment (Rails 3.0.5)
irb(main):007:0> v = User.find :first
User Load (0.6ms) SELECT 'users'.* FROM 'users' LIMIT 1
=> #<User id: 1, username: "Senki", password: "titok", email: "senki@mail.
    bme.hu", created_at: "2011-10-04 11:20:00", updated_at: "2011-10-04
    11:20:00">
irb(main):008:0> v.username
=> "Senki"

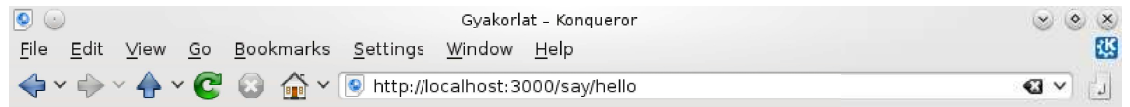
```

Hivatkozások

- [1] Sam Ruby, Dave Thomas, and David Heinemeier Hansson et al. *Agile Web Development with Rails*, volume Third Edition. The Pragmatic Bookshelf, 2009 Mar.



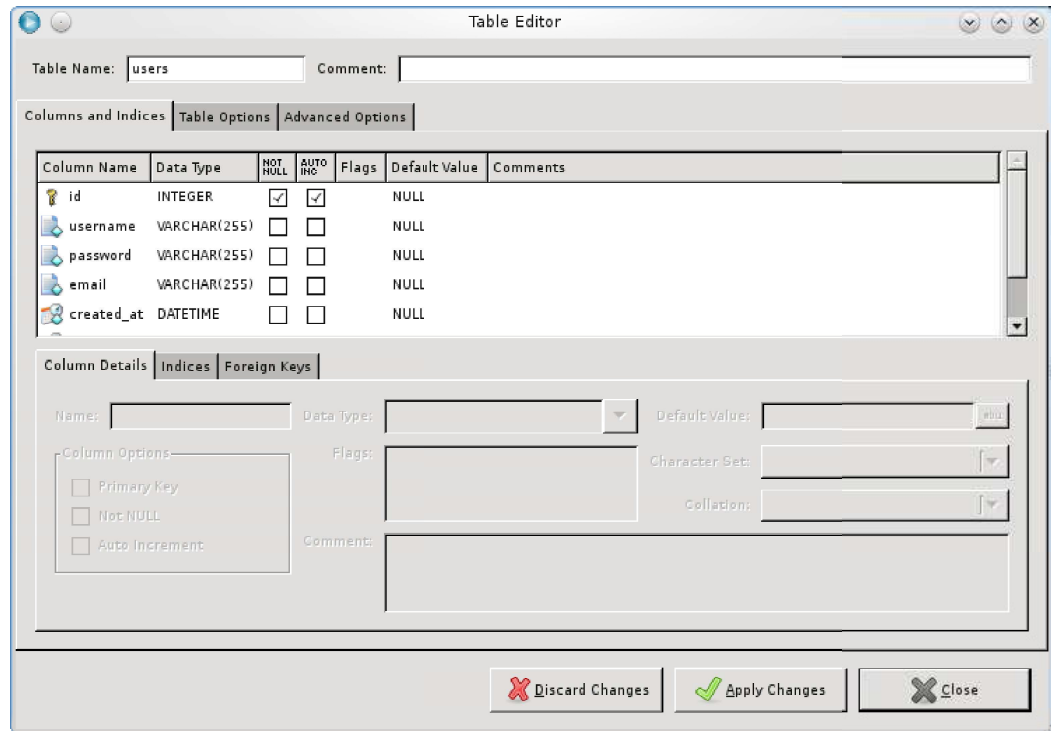
1. ábra. A beágyazott webszerver elérése



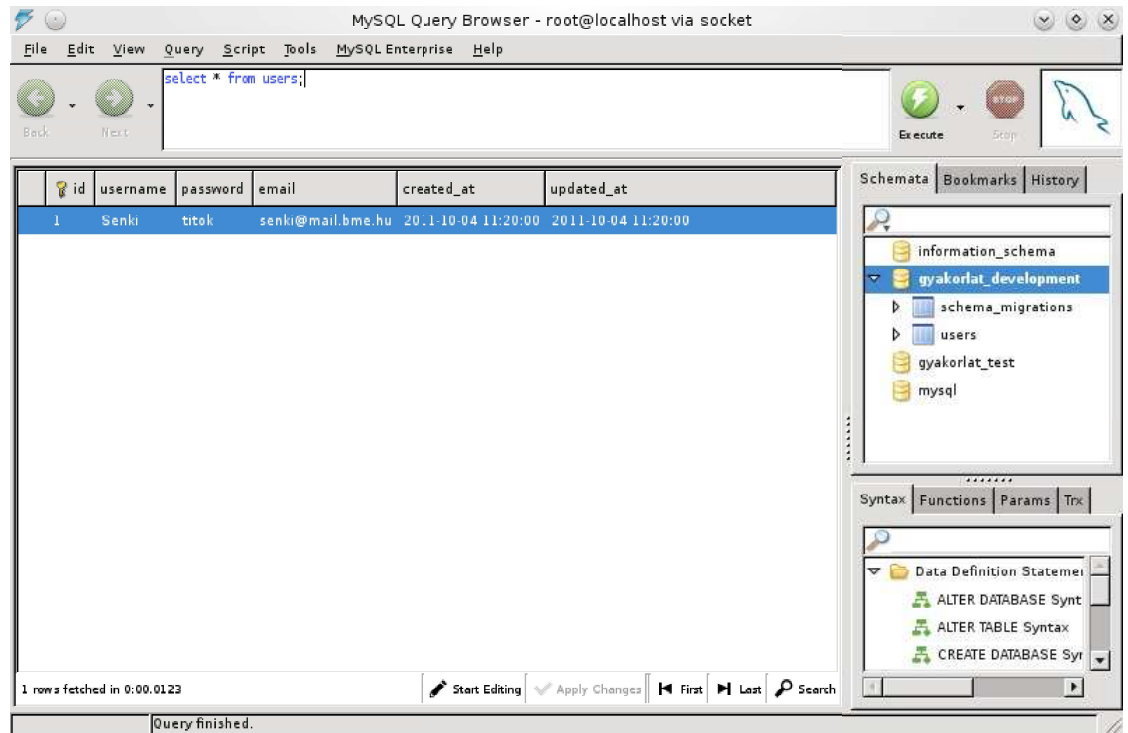
Hello, world!

2011-10-04 13:04:53 +0200

2. ábra. A hello akció megjelenítve



3. ábra. A migráció által létrehozott tábla



4. ábra. Az elmetett rekord