

A Rails keretrendszer Gyakorlat

Kovács Gábor

2012. február 29.

1. Ruby telepítése, Gemek

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 és néhány további komponens 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:

```
kovacsg@rails:~$ sudo bash
[sudo] password for kovacsg:
root@rails:/home/kovacsg# su
root@rails:/home/kovacsg# apt-get install ruby1.9.1 ruby1.9.1-dev ri1.9.1
```

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.

```

Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libruby1.9.1 libyaml-0-2
Suggested packages:
  ruby1.9.1-examples
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 0 not upgraded.
Need to get 9,270 kB of archives.
After this operation, 64.9 MB of additional disk space will be used.
Do you want to continue [Y/n]?
Get:1 http://ftp.debian.org/debian/ squeeze/main libyaml-0-2 amd64
    0.1.3-1 [54.6 kB]
Get:2 http://ftp.hu.debian.org/debian/ squeeze/main libruby1.9.1 amd64
    1.9.2.0-2 [4,517 kB]
Get:3 http://ftp.hu.debian.org/debian/ squeeze/main ruby1.9.1 amd64
    1.9.2.0-2 [754 kB]
Get:4 http://ftp.hu.debian.org/debian/ squeeze/main ri1.9.1 all 1.9.2.0-2
    [2,093 kB]
Get:5 http://ftp.hu.debian.org/debian/ squeeze/main ruby1.9.1-dev amd64
    1.9.2.0-2 [1,852 kB]
Fetched 9,270 kB in 6s (1,434 kB/s)
Selecting previously deselected package libyaml-0-2.
(Reading database ... 98605 files and directories currently installed.)
Unpacking libyaml-0-2 (from .../libyaml-0-2_0.1.3-1_amd64.deb) ...
Selecting previously deselected package libruby1.9.1.
Unpacking libruby1.9.1 (from .../libruby1.9.1_1.9.2.0-2_amd64.deb) ...
Selecting previously deselected package ruby1.9.1.
Unpacking ruby1.9.1 (from .../ruby1.9.1_1.9.2.0-2_amd64.deb) ...
Selecting previously deselected package ri1.9.1.
Unpacking ri1.9.1 (from .../ri1.9.1_1.9.2.0-2_all.deb) ...
Selecting previously deselected package ruby1.9.1-dev.
Unpacking ruby1.9.1-dev (from .../ruby1.9.1-dev_1.9.2.0-2_amd64.deb) ...
Processing triggers for man-db ...
Setting up libyaml-0-2 (0.1.3-1) ...
Setting up libruby1.9.1 (1.9.2.0-2) ...
Setting up ruby1.9.1 (1.9.2.0-2) ...
Setting up ri1.9.1 (1.9.2.0-2) ...
Setting up ruby1.9.1-dev (1.9.2.0-2) ...

```

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

```

root@debian:/home/kovacs# ruby --version
ruby 1.9.2p290 (2011-07-09 revision 32553) [x86_64-linux]
root@debian:/home/kovacs# 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@rails:/home/kovacs# 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 i18n-0.6.0
Successfully installed multi_json-1.1.0
Successfully installed activesupport-3.2.1
Successfully installed builder-3.0.0
Successfully installed activemodel-3.2.1
Successfully installed rack-1.4.1
Successfully installed rack-cache-1.1
Successfully installed rack-test-0.6.1
Successfully installed journey-1.0.3
Successfully installed hike-1.2.1
Successfully installed tilt-1.3.3
Successfully installed sprockets-2.1.2
Successfully installed erubis-2.7.0
Successfully installed actionpack-3.2.1
Successfully installed arel-3.0.2
Successfully installed tzinfo-0.3.31
Successfully installed activerecord-3.2.1
Successfully installed activerecord-3.2.1
Successfully installed mime-types-1.17.2
Successfully installed polyglot-0.3.3
Successfully installed treetop-1.4.10
Successfully installed mail-2.4.1
Successfully installed actionmailer-3.2.1
Successfully installed rake-0.9.2.2
Successfully installed thor-0.14.6
Successfully installed rack-ssl-1.3.2
Successfully installed json-1.6.5
Successfully installed rdoc-3.12
Successfully installed railties-3.2.1
Successfully installed bundler-1.0.22
Successfully installed rails-3.2.1
31 gems installed
Installing ri documentation for i18n-0.6.0...
Installing ri documentation for multi_json-1.1.0...
Installing ri documentation for activesupport-3.2.1...
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 activemodel-3.2.1...
Installing ri documentation for rack-1.4.1...
Installing ri documentation for rack-cache-1.1...
Installing ri documentation for rack-test-0.6.1...
Installing ri documentation for journey-1.0.3...
Installing ri documentation for hike-1.2.1...
Installing ri documentation for tilt-1.3.3...
Installing ri documentation for sprockets-2.1.2...
Installing ri documentation for erubis-2.7.0...
Installing ri documentation for actionpack-3.2.1...
Installing ri documentation for arel-3.0.2...
Installing ri documentation for tzinfo-0.3.31...
Installing ri documentation for activerecord-3.2.1...
Installing ri documentation for activerecord-3.2.1...
Installing ri documentation for mime-types-1.17.2...
Installing ri documentation for polyglot-0.3.3...
Installing ri documentation for treetop-1.4.10...
Installing ri documentation for mail-2.4.1...
Installing ri documentation for actionmailer-3.2.1...
Installing ri documentation for rake-0.9.2.2...
Installing ri documentation for thor-0.14.6...
Installing ri documentation for rack-ssl-1.3.2...

```

```

Installing ri documentation for json-1.6.5...
Installing ri documentation for rdoc-3.12...
Installing ri documentation for railties-3.2.1...
Installing ri documentation for bundler-1.0.22...
Installing ri documentation for rails-3.2.1...
Installing RDoc documentation for i18n-0.6.0...
Installing RDoc documentation for multi_json-1.1.0...
Installing RDoc documentation for activesupport-3.2.1...
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 activemodel-3.2.1...
Installing RDoc documentation for rack-1.4.1...
Installing RDoc documentation for rack-cache-1.1...
Installing RDoc documentation for rack-test-0.6.1...
Installing RDoc documentation for journey-1.0.3...
Installing RDoc documentation for hike-1.2.1...
Installing RDoc documentation for tilt-1.3.3...
Installing RDoc documentation for sprockets-2.1.2...
Installing RDoc documentation for erubis-2.7.0...
Installing RDoc documentation for actionpack-3.2.1...
Installing RDoc documentation for arel-3.0.2...
Installing RDoc documentation for tzinfo-0.3.31...
Installing RDoc documentation for activerecord-3.2.1...
Installing RDoc documentation for activeresource-3.2.1...
^|=Installing RDoc documentation for mime-types-1.17.2...
Installing RDoc documentation for polyglot-0.3.3...
Installing RDoc documentation for treetop-1.4.10...
Installing RDoc documentation for mail-2.4.1...
Installing RDoc documentation for actionmailer-3.2.1...
Installing RDoc documentation for rake-0.9.2.2...
Installing RDoc documentation for thor-0.14.6...
Installing RDoc documentation for rack-ssl-1.3.2...
Installing RDoc documentation for json-1.6.5...
Installing RDoc documentation for rdoc-3.12...
Installing RDoc documentation for railties-3.2.1...
Installing RDoc documentation for bundler-1.0.22...
Installing RDoc documentation for rails-3.2.1...

```

Railshez egy opcionális, ám az éles rendszerek szempontjából annál hasznosabb komponens az Apache webserverral való integrációt lehetővé tevő Passenger plugin.

```

root@rails:/home/kovacs# gem install passenger
Building native extensions. This could take a while...
Successfully installed fastthread-1.0.7
Successfully installed daemon_controller-1.0.0
Successfully installed passenger-3.0.11
3 gems installed
Installing ri documentation for fastthread-1.0.7...
Installing ri documentation for daemon_controller-1.0.0...
Installing ri documentation for passenger-3.0.11...
Installing RDoc documentation for fastthread-1.0.7...
Installing RDoc documentation for daemon_controller-1.0.0...
Installing RDoc documentation for passenger-3.0.11...

```

Ezután telepíthetjük először az SQLite, majd a MySQL adapterének Ruby API-ját. Ez két lépésből áll, először a operációs rendszertől függő fájlok fordításához szükséges forrásokat (C és C++ headereket) telepítjük.

```

root@rails:/home/kovacs# apt-get install libsqlite3-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  sqlite3-doc
The following NEW packages will be installed:
  libsqlite3-dev
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 498 kB of archives.
After this operation, 1,229 kB of additional disk space will be used.
Get:1 http://ftp.hu.debian.org/debian/ squeeze/main libsqlite3-dev amd64
    3.7.3-1 [498 kB]
Fetched 498 kB in 5s (97.8 kB/s)
Selecting previously deselected package libsqlite3-dev.
(Reading database ... 110338 files and directories currently installed.)
Unpacking libsqlite3-dev (from .../libsqlite3-dev_3.7.3-1_amd64.deb) ...
Setting up libsqlite3-dev (3.7.3-1) ...
root@rails:/home/kovacs# apt-get install libmysqlclient-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
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 0 not upgraded.
Need to get 3,406 kB of archives.
After this operation, 10.3 MB of additional disk space will be used.
Do you want to continue [Y/n]?
Get:1 http://ftp.hu.debian.org/debian/ squeeze/main zlib1g-dev amd64
    1:1.2.3.4.dfsg-3 [192 kB]
Get:2 http://ftp.hu.debian.org/debian/ squeeze/main libmysqlclient-dev amd64
    5.1.49-3 [3,214 kB]
Fetched 3,406 kB in 5s (638 kB/s)
Selecting previously deselected package zlib1g-dev.
(Reading database ... 110251 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_5.1.49-3_amd64.deb
) ...
Processing triggers for man-db ...
Setting up zlib1g-dev (1:1.2.3.4.dfsg-3) ...
Setting up libmysqlclient-dev (5.1.49-3) ...

```

Ezután telepíthetjük magukat az adatbáziskezelő-adaptereket.

```

root@rails:/home/kovacs# gem install mysql2
Building native extensions. This could take a while...
Successfully installed mysql2-0.3.11
1 gem installed
Installing ri documentation for mysql2-0.3.11...
Installing RDoc documentation for mysql2-0.3.11...
root@rails:/home/kovacs# gem install sqlite3
Building native extensions. This could take a while...
Successfully installed sqlite3-1.3.5
1 gem installed
Installing ri documentation for sqlite3-1.3.5...
Installing RDoc documentation for sqlite3-1.3.5...

```

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@rails:/home/kovacs# gem install execjs
Successfully installed execjs-1.3.0
1 gem installed
Installing ri documentation for execjs-1.3.0...
Installing RDoc documentation for execjs-1.3.0...
root@rails:/home/kovacs# gem install therubyracer
Building native extensions. This could take a while...
Successfully installed therubyracer-0.9.10
1 gem installed
Installing ri documentation for therubyracer-0.9.10...
Installing RDoc documentation for therubyracer-0.9.10...
```

A Rails rendszerünk ezután az alábbi Ruby függvénykönyvtárakból áll

```
kovacs@rails:~$ gem list

*** LOCAL GEMS ***

actionmailer (3.2.1)
actionpack (3.2.1)
activemodel (3.2.1)
activerecord (3.2.1)
activeresource (3.2.1)
activesupport (3.2.1)
arel (3.0.2)
builder (3.0.0)
bundler (1.0.22)
coffee-rails (3.2.2)
coffee-script (2.2.0)
coffee-script-source (1.2.0)
daemon_controller (1.0.0)
erubis (2.7.0)
execjs (1.3.0)
fastthread (1.0.7)
hike (1.2.1)
i18n (0.6.0)
journey (1.0.3)
jquery-rails (2.0.1)
json (1.6.5)
libv8 (3.3.10.4 x86_64-linux)
mail (2.4.1)
mime-types (1.17.2)
multi_json (1.1.0)
mysql2 (0.3.11)
passenger (3.0.11)
polyglot (0.3.3)
rack (1.4.1)
rack-cache (1.1)
rack-ssl (1.3.2)
rack-test (0.6.1)
rails (3.2.1)
railties (3.2.1)
rake (0.9.2.2, 0.9.2, 0.8.7)
rdoc (3.12)
sass (3.1.15)
sass-rails (3.2.4)
sprockets (2.1.2)
```

```
sqlite3 (1.3.5)
therubyracer (0.9.10)
thor (0.14.6)
tilt (1.3.3)
treetop (1.4.10)
tzinfo (0.3.31)
uglifyer (1.2.3)
```

2. Rails alkalmazás létrehozása

A Rails használatához elérhetővé kell tennünk a rails szkriptet, amely Debian/Ubuntu Linux és 1.9.1-es ruby esetén a `/var/lib/gems/1.9.1/bin/` könyvtárba kerül.

```
kovacsg@rails:~$ export PATH=$PATH:/var/lib/gems/1.9.1/bin/
```

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 `sqlite3`-ot mint alapértelmezett adatbáziskezelőt és a `mysql`-t fogjuk használni.

```
kovacsg@rails:~$ 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/sqlserver/
  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.

You can specify extra command-line arguments to be used every time
'rails new' runs in the .railsrc configuration file in your home
directory.

Note that the arguments specified in the .railsrc file don't affect the
defaults values shown above in this help message.

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.

A parancs végén automatikusan lefut a Rails keretrendszer egy másik parancsa, a `bundle`, amely az `install` opció hatására összeszedi a gemek közül azokat, amelyekre az alkalmazásunknak szüksége lesz. A konzolon kiadott `bundle install` paranccsal később telepíthetjük az összes az esetlegesen hiányzó vagy nem megfelelő verziójú Ruby függvénykönyvtárat, a `bundle update` paranccsal pedig az éppen használni kívánt verziót tudunk betölteni az egyes gemekből.

```
kovacsg@rails:~$ rails new gyakorlat2
create
create  README.rdoc
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/javascripts
create vendor/assets/javascripts/.gitkeep
create vendor/assets/stylesheets
create vendor/assets/stylesheets/.gitkeep
create vendor/plugins
create vendor/plugins/.gitkeep
run bundle install
```

```

Fetching source index for https://rubygems.org/
Could not reach rubygems repository https://rubygems.org/
Using rake (0.9.2.2)
Using i18n (0.6.0)
Using multi_json (1.1.0)
Using activesupport (3.2.1)
Using builder (3.0.0)
Using activemodel (3.2.1)
Using erubis (2.7.0)
Using journey (1.0.3)
Using rack (1.4.1)
Using rack-cache (1.1)
Using rack-test (0.6.1)
Using hike (1.2.1)
Using tilt (1.3.3)
Using sprockets (2.1.2)
Using actionpack (3.2.1)
Using mime-types (1.17.2)
Using polyglot (0.3.3)
Using treetop (1.4.10)
Using mail (2.4.1)
Using actionmailer (3.2.1)
Using arel (3.0.2)
Using tzinfo (0.3.31)
Using activerecord (3.2.1)
Using activeresource (3.2.1)
Using bundler (1.0.22)
Using coffee-script-source (1.2.0)
Using execjs (1.3.0)
Using coffee-script (2.2.0)
Using rack-ssl (1.3.2)
Using json (1.6.5)
Using rdoc (3.12)
Using thor (0.14.6)
Using railties (3.2.1)
Using coffee-rails (3.2.2)
Using jquery-rails (2.0.1)
Using rails (3.2.1)
Using sass (3.1.15)
Using sass-rails (3.2.4)
Using sqlite3 (1.3.5)
Using uglifier (1.2.3)
Your bundle is complete! Use 'bundle show [gemname]' to see where a bundled
gem is installed.

```

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 fájljait, 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 környezetben vagy a már említett lib és vendor könyvtárakban érhetők el.

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 szűrjük be:

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

3. Web- és adatbázisszerver kapcsolat beállítása

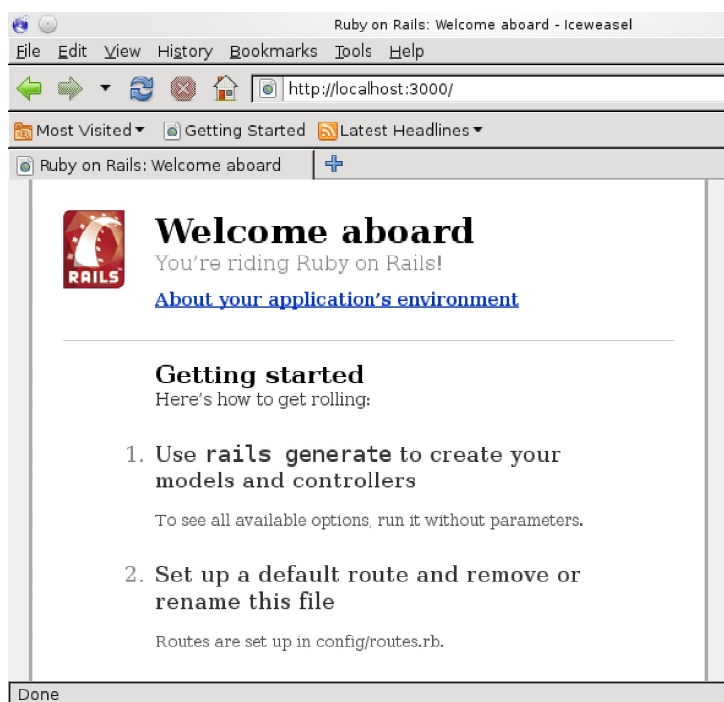
A Rails keretrendszer webszerverek számára állít elő dinamikus tartalmat adatbázisbeli adatok alapján. A gyakorlat során kétféle webservert és adatbáziskezelőt vizsgáltunk meg.

A beépített webservert a rails server opciójával indíthatjuk el a Rails alkalmazásunk tetszőleges könyvtárából. Ha a szerveret 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@rails:~$ rails server
=> Booting WEBrick
=> Rails 3.2.1 application starting in development on http://0.0.0.0:3000
=> Call with -d to detach
=> Ctrl-C to shutdown server
[2012-02-29 12:44:37] INFO WEBrick 1.3.1
[2012-02-29 12:44:37] INFO ruby 1.9.2 (2010-08-18) [x86_64-linux]
[2012-02-29 12:44:37] INFO WEBrick::HTTPServer#start: pid=3818 port=3000
```

A Rails a már említett Passenger plugin segítségével illeszhető Apache webservertre. Ehhez a következő parancsot kell kiadnunk. A parancs lefordítja, majd telepíti az Apache webservert Rails modulját, a konzolra kiírja az Apache2 konfigurációs állományába beírandó Rails specifikus három sort, és a Rails alkalmazásunkhoz való hozzáféréshez szükséges VirtualHost beállításokat. A Passenger alapértelmezés szerint a Rails éles környezetével működik együtt, ha ezt módosítani szeretnénk, akkor a RailsEnv development sort kell elhelyeznünk az Apache konfigurációs állományába a fejlesztői környezet eléréséhez.

```
passenger-install-apache2-module
```



1. ábra. A beágyazott webszerver elérése

A Rails kényelmesebb hozzáférése végett módosíthatjuk a helyi gépen doménnévtáblát (Windowson `C:\windows\system32\drivers\etc\hosts`, Linuxon `/etc/hosts`) a következő bejegyzés hozzáadva, amely minden, a `gyakorlat.com`, illetve `www.gyakorlat.com` címre küldött kérést a helyi gépen kezeltet le.

```
127.0.0.1      gyakorlat.com www.gyakorlat.com
```

Az Apache virtuális hoszt beállításai ezek alapján a következő lehet:

```
<VirtualHost *:80>
  ServerName www.gyakorlat.com
  ServerAdmin admin@gyakorlat.com
```

```

DocumentRoot /Users/kovacs/gyakorlat2/public
ServerSignature On

CustomLog /private/var/log/apache2/gyakorlat_access.log combined
ErrorLog /private/var/log/apache2/gyakorlat_error.log
LogLevel info

<Directory />
  Options FollowSymLinks
  AllowOverride None
</Directory>
<Directory /Users/kovacs/gyakorlat2/public/>
  Options Indexes FollowSymLinks MultiViews
  AllowOverride None
  Order allow,deny
  Allow from all
</Directory>

```

Ezután egy böngészőbe beírva a `www.gyakorlat.com` címet a Rails alkalmazásunk fog megjelenni!

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 (`rails new gyakorlat -d mysql`) a `mysql2` adaptert hasz-

náljuk. A különbség a `Gemfile`-ban és a `database.yml`-ben jelentkezik. Fontos opció az alapértelmezett karakterkódolás megadása, ami, ha lehet hagyjunk változatlanul `utf8` értéken. A `database` opció a séma nevét tartalmazza értéként. A felhasználónév és jelszó megadása kötelező. A kapcsolat lehet `socket` vagy `host` és `port` alapon megadott.

Távoli gépen található MySQL esetén szükséges a Rails felhasználó hozzáférési jogosultságainak beállítása, amelyet a MySQL `mysql` adatbázisának `user` táblájában kell megtennünk. A Rails szerverének IP címére engedélyeznünk kell a Rails adatbázis-konfigurációs állományában megadott felhasználónévvel és jelszóval azonosított felhasználó számára a `select`, `update`, `insert`, `delete`, `create`, `alter`, `drop` és `index` jogosultságokat.

Az adatbáziskapcsolatnak a szerver elindítása előtt léteznie kell. Ez SQLite esetén nem probléma, hiszen a fejlesztői adatbázisfájl automatikusan létrejön, MySQL esetén viszont a sémákat explicite létre kell hoznunk. Ebben a `rake` parancs lesz segítségünkre, amelyet a Rails keretrendszerhez kapcsolódó műveletekre használunk. A műveletek listája alább látható.

```
bash-3.2$ rake -T
rake about                # List versions of all Rails frameworks and the env
...
rake assets:clean        # Remove compiled assets
rake assets:precompile   # Compile all the assets named in config/assets.pre
...
rake db:create           # Create the database from config/database.yml for
...
rake db:drop             # Drops the database for the current Rails.env (use
...
rake db:fixtures:load    # Load fixtures into the current environment's data
...
rake db:migrate          # Migrate the database (options: VERSION=x, VERBOSE
...
rake db:migrate:status   # Display status of migrations
rake db:rollback         # Rolls the schema back to the previous version (sp
...
rake db:schema:dump      # Create a db/schema.rb file that can be portably u
...
rake db:schema:load      # Load a schema.rb file into the database
rake db:seed             # Load the seed data from db/seeds.rb
rake db:setup            # Create the database, load the schema, and initial
...
rake db:structure:dump   # Dump the database structure to db/structure.sql.
...
rake db:version          # Retrieves the current schema version number
rake doc:app             # Generate docs for the app — also available doc:r
...
rake log:clear           # Truncates all *.log files in log/ to zero bytes
rake middleware          # Prints out your Rack middleware stack
rake notes               # Enumerate all annotations (use notes:optimize, :f
...
rake notes:custom        # Enumerate a custom annotation, specify with ANNOT
...
rake rails:template      # Applies the template supplied by LOCATION=(/path
/...
rake rails:update        # Update configs and some other initially generated
```

```

rake routes          # Print out all defined routes in match order, with
rake secret         # Generate a cryptographically secure secret key (t
rake stats          # Report code statistics (KLOCs, etc) from the appl
rake test           # Runs test:units, test:functionals, test:integrati
rake test:recent    # Run tests for recenttest:prepare / Test recent ch
rake test:single    # Run tests for singletest:prepare
rake test:uncommitted # Run tests for uncommittedtest:prepare / Test chan
rake time:zones:all # Displays all time zones, also available: time:zon
rake tmp:clear      # Clear session, cache, and socket files from tmp/
rake tmp:create     # Creates tmp directories for sessions, cache, sock

```

Az adatbázis létrehozása következő paranccsal történhet meg, amely a development és a test környezetekhez hozza létre egy üres sémát.

```
bash-3.2$ rake db:create
```

4. Bevezetés a Rails használatába

Az adatbáziskapcsolat-leíró mellett a másik fontos konfigurációs fájlunk a `routes.rb`. Ez azt adja meg, hogy milyen struktúrájú legyen az URL, amivel elérjük a Rails alkalmazásunk egyes funkcióit. Az alábbi kódrészlet a legáltalánosabb beállítást tartalmazza. A webserverver 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.

Hozzunk létre egy új controllert a rails parancs `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 akciókat. E parancs négy Ruby forrásfajlt 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.

```

kovacsg@rails:~/gyakorlat$ rails generate controller say hello
  create  app/controllers/say_controller.rb
  route  get "say/hello"
  invoke erb
  create  app/views/say
  create  app/views/say/hello.html.erb
  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 kiírásával.

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

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

```

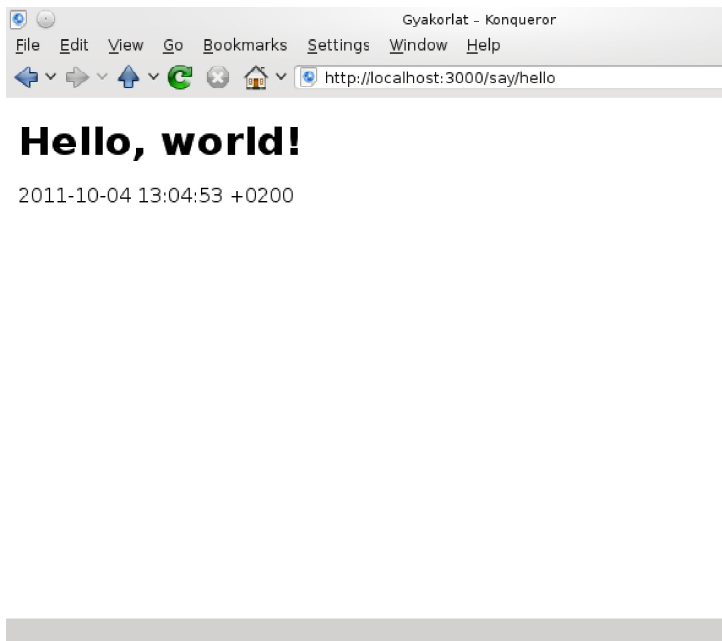
class SayController < ApplicationController
  def hello
    @time=Time.now
  end
end

```

A nézetben pedig csak hivatkozunk a Controller példányváltozóra.

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

A controllernek és akciónak megfelelő megjelenített oldalt a 2. ábra mutatja.



2. ábra. A hello akció megjelenítve

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>
2012-02-29 13:04:53 +0200

</body>
</html>

```

A Rails MVC filozófiájának harmadik eleme a modell, amelyet szintén a `rails generate` 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 egységteszteket tartalmazó osztályt és egy tesztadatokat tartalmazó YAML fájlt.

```

kovacs@rails:~/gyakorlat/app/controllers$ rails generate model User
invoke active_record
create db/migrate/20120229123502_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 az `email`, `password` és `username` azonosítókkal, és egy boolean típusú attribútumot, amely azt adja meg, hogy adminisztrátor felhasználóról van-e szó. A változás most egy a `create_table` metódus által létrehozandó tábla hét attribútummal.

```

class CreateUsers < ActiveRecord::Migration
  def change
    create_table :users do |t|
      t.string :username
      t.string :password
      t.string :email
      t.boolean :admin
      t.timestamps
    end
  end
end

```

Ha a Rails alkalmazás létrehozásuk megadtuk volna a `mysql` adatbázis opciót, akkor először létre kell hoznunk a megfelelő táblát, majd utána elvégeztetjük a tábla struktúrájának módosítását. Ezekben a `rake` parancs

nyújt segítséget. A `rake db:create` létrehozza a táblákat, ha még nem tettük volna meg, a `rake db:migrate` módosítja a séma struktúráját.

```
kovacs@rails:~/gyakorlat/db/migrate$ rake db:migrate
== CreateUsers: migrating
-----
-- create_table(:users)
--> 0.0668s
== CreateUsers: migrated (0.0670s)
-----
```

Az adatbáziskezelővel ezután ellenőrizhetjük, hogy a táblánk valóban létrejött.

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.