

A Rails keretrendszer Gyakorlat

Kovács Gábor

2023. október 3.

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/MariaDB adatbázis kezelő lesz. Ezt az `apt-get install mariadb-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 környezeten az egyszerűség kedvéért üres jelszót fogunk használni.

A Rails adatbázis adapterei és néhány további komponens fordításához szükségünk van C és C++ fordítóra, a GIT verziókezelőre, valamint a fejlesztői függvénykönyvtárakra ³, amelyeket az `apt-get install build-essential`, `apt-get install git`, `apt-get install gcc` és `apt-get install g++` parancsok konzolon való kiadásával telepíthetünk.

A gyakorlat folyamán illesztjük az alkalmazásunkat egy Apache2 webserververhez is, amelyet az `apt-get install apache2` parancs konzolon történő kiadásával telepíthetünk.

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.

³A Rails egyik függvénykönyvtára miatt szükségünk lesz még a `zlib1g` csomagra is.

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. A gyakorlaton a Ruby 3.1-s verzióját használjuk. A C függvénykönyvtárakhoz való illesztéshez szükségünk lesz a Ruby header fájlokra is, ezért telepítjük azok Linux csomagját is. Ezeket a következő paranccsal telepíthetjük rendszergazdaként:

```
kovacs@debian:~# ruby
bash: ruby: command not found
kovacs@debian:~# sudo bash
[sudo] password for kovacs:
root@debian:~/home/kovacs# su -
root@debian:~# apt-get install ruby ruby-dev ri rdoc
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'ruby' instead of 'rdoc'
The following additional packages will be installed:
  fonts-lato libruby libruby3.1 rake ruby-net-telnet ruby-rubygems ruby-sdbm
  ruby-webrick ruby-xmlrpc ruby3.1 ruby3.1-dev ruby3.1-doc
  rubygems-integration
Suggested packages:
  bundler
The following NEW packages will be installed:
  fonts-lato libruby libruby3.1 rake ri ruby ruby-dev ruby-net-telnet
  ruby-rubygems ruby-sdbm ruby-webrick ruby-xmlrpc ruby3.1 ruby3.1-dev
  ruby3.1-doc rubygems-integration
0 upgraded, 16 newly installed, 0 to remove and 16 not upgraded.
Need to get 0 B/12.7 MB of archives.
After this operation, 62.0 MB of additional disk space will be used.
Do you want to continue? [Y/n]
Selecting previously unselected package fonts-lato.
(Reading database ... 227398 files and directories currently installed.)
Preparing to unpack .../00-fonts-lato_2.0-2.1_all.deb ...
Unpacking fonts-lato (2.0-2.1) ...
Selecting previously unselected package rubygems-integration.
Preparing to unpack .../01-rubygems-integration_1.18_all.deb ...
Unpacking rubygems-integration (1.18) ...
Selecting previously unselected package ruby3.1.
Preparing to unpack .../02-ruby3.1_3.1.2-7_amd64.deb ...
Unpacking ruby3.1 (3.1.2-7) ...
Selecting previously unselected package ruby-rubygems.
Preparing to unpack .../03-ruby-rubygems_3.3.15-2_all.deb ...
Unpacking ruby-rubygems (3.3.15-2) ...
Selecting previously unselected package ruby.
Preparing to unpack .../04-ruby_1%3a3.1_amd64.deb ...
Unpacking ruby (1:3.1) ...
Selecting previously unselected package rake.
Preparing to unpack .../05-rake_13.0.6-3_all.deb ...
Unpacking rake (13.0.6-3) ...
Selecting previously unselected package ruby-net-telnet.
Preparing to unpack .../06-ruby-net-telnet_0.2.0-1_all.deb ...
Unpacking ruby-net-telnet (0.2.0-1) ...
Selecting previously unselected package ruby-webrick.
Preparing to unpack .../07-ruby-webrick_1.8.1-1_all.deb ...
Unpacking ruby-webrick (1.8.1-1) ...
Selecting previously unselected package ruby-xmlrpc.
Preparing to unpack .../08-ruby-xmlrpc_0.3.2-2_all.deb ...
Unpacking ruby-xmlrpc (0.3.2-2) ...
Selecting previously unselected package ruby-sdbm:amd64.
```

```

Preparing to unpack .../09-ruby-sdbm_1.0.0-5+b1_amd64.deb ...
Unpacking ruby-sdbm:amd64 (1.0.0-5+b1) ...
Selecting previously unselected package libruby3.1:amd64.
Preparing to unpack .../10-libruby3.1_3.1.2-7_amd64.deb ...
Unpacking libruby3.1:amd64 (3.1.2-7) ...
Selecting previously unselected package libruby:amd64.
Preparing to unpack .../11-libruby_1%3a3.1_amd64.deb ...
Unpacking libruby:amd64 (1:3.1) ...
Selecting previously unselected package ruby3.1-doc.
Preparing to unpack .../12-ruby3.1-doc_3.1.2-7_all.deb ...
Unpacking ruby3.1-doc (3.1.2-7) ...
Selecting previously unselected package ri.
Preparing to unpack .../13-ri_1%3a3.1_all.deb ...
Unpacking ri (1:3.1) ...
Selecting previously unselected package ruby3.1-dev:amd64.
Preparing to unpack .../14-ruby3.1-dev_3.1.2-7_amd64.deb ...
Unpacking ruby3.1-dev:amd64 (3.1.2-7) ...
Selecting previously unselected package ruby-dev:amd64.
Preparing to unpack .../15-ruby-dev_1%3a3.1_amd64.deb ...
Unpacking ruby-dev:amd64 (1:3.1) ...
Setting up fonts-lato (2.0-2.1) ...
Setting up rubygems-integration (1.18) ...
Setting up ruby3.1-doc (3.1.2-7) ...
Setting up ruby-net-telnet (0.2.0-1) ...
Setting up ruby-webrick (1.8.1-1) ...
Setting up ruby-xmlrpc (0.3.2-2) ...
Setting up ruby-rubygems (3.3.15-2) ...
Setting up ruby3.1 (3.1.2-7) ...
Setting up ri (1:3.1) ...
Setting up ruby (1:3.1) ...
Setting up rake (13.0.6-3) ...
Setting up libruby3.1:amd64 (3.1.2-7) ...
Setting up libruby:amd64 (1:3.1) ...
Setting up ruby3.1-dev:amd64 (3.1.2-7) ...
Setting up ruby-sdbm:amd64 (1.0.0-5+b1) ...
Setting up ruby-dev:amd64 (1:3.1) ...
Processing triggers for man-db (2.11.2-2) ...
Processing triggers for fontconfig (2.14.1-4) ...
Processing triggers for libc-bin (2.36-9+deb12u1) ...

```

Ezután ellenőrizzük, hogy elérhető-e a ruby értelmező és a gem Ruby csomagkezelő, és nézzük meg az előre telepített Ruby API-k listáját.

```

root@debian:~# ruby --version
ruby 3.1.2 p20 (2022-04-12 revision 4491bb740a) [x86_64-linux-gnu]
root@debian:~# irb
irb(main):001:0>

```

A következő lépés a Ruby dokumentációgeneráló függvénykönyvtárának telepítése, amit a gem Ruby csomagkezelővel teszünk meg. A lépést kihagyva a Rails keretrendszer dokumentációjának telepítése sikertelen lenne, ami mindazonáltal nem nagy tragédia lévén annak telepítését amúgy is előszere-ttel kikapcsoljuk a gem --no-rdoc --no-ri kapcsolóival. Először azonban hibára futottunk, mert az egyik szükséges C fejlesztői függvénykönyvtár nem volt elérhető.

```

root@debian:~# gem install rails
Fetching concurrent-ruby-1.2.2.gem

```

```

Fetching minitest -5.20.0.gem
Fetching zeitwerk -2.6.12.gem
Fetching i18n -1.14.1.gem
Fetching thor -1.2.2.gem
Fetching method_source -1.0.0.gem
Fetching tzinfo -2.0.6.gem
Fetching activesupport -7.0.8.gem
Fetching nokogiri -1.15.4-x86_64-linux.gem
Fetching crass -1.0.6.gem
Fetching loofah -2.21.3.gem
Fetching rails-html-sanitizer -1.6.0.gem
Fetching rails-dom-testing -2.2.0.gem
Fetching rack -2.2.8.gem
Fetching rack-test -2.1.0.gem
Fetching erubi -1.12.0.gem
Fetching builder -3.2.4.gem
Fetching actionview -7.0.8.gem
Fetching actionpack -7.0.8.gem
Fetching raities -7.0.8.gem
Fetching mini_mime -1.1.5.gem
Fetching marcel -1.0.2.gem
Fetching activemodel -7.0.8.gem
Fetching activerecord -7.0.8.gem
Fetching globalid -1.2.1.gem
Fetching activejob -7.0.8.gem
Fetching activestorage -7.0.8.gem
Fetching actiontext -7.0.8.gem
Fetching mail -2.8.1.gem
Fetching actionmailer -7.0.8.gem
Fetching actionmailbox -7.0.8.gem
Fetching websocket-extensions -0.1.5.gem
Fetching websocket-driver -0.7.6.gem
Fetching nio4r -2.5.9.gem
Fetching actioncable -7.0.8.gem
Fetching rails -7.0.8.gem
Successfully installed zeitwerk -2.6.12
Successfully installed thor -1.2.2
Successfully installed method_source -1.0.0
Successfully installed concurrent-ruby -1.2.2
Successfully installed tzinfo -2.0.6
Successfully installed minitest -5.20.0
Successfully installed i18n -1.14.1
Successfully installed activesupport -7.0.8
Successfully installed nokogiri -1.15.4-x86_64-linux
Successfully installed crass -1.0.6
Successfully installed loofah -2.21.3
Successfully installed rails-html-sanitizer -1.6.0
Successfully installed rails-dom-testing -2.2.0
Successfully installed rack -2.2.8
Successfully installed rack-test -2.1.0
Successfully installed erubi -1.12.0
Successfully installed builder -3.2.4
Successfully installed actionview -7.0.8
Successfully installed actionpack -7.0.8
Successfully installed raities -7.0.8
Successfully installed mini_mime -1.1.5
Successfully installed marcel -1.0.2
Successfully installed activemodel -7.0.8
Successfully installed activerecord -7.0.8
Successfully installed globalid -1.2.1
Successfully installed activejob -7.0.8
Successfully installed activestorage -7.0.8

```

```
Successfully installed actiontext-7.0.8
Successfully installed mail-2.8.1
Successfully installed actionmailer-7.0.8
Successfully installed actionmailbox-7.0.8
Successfully installed websocket-extensions-0.1.5
Building native extensions. This could take a while...
Successfully installed websocket-driver-0.7.6
Building native extensions. This could take a while...
Successfully installed nio4r-2.5.9
Successfully installed actioncable-7.0.8
Successfully installed rails-7.0.8
Parsing documentation for zeitwerk-2.6.12
Installing ri documentation for zeitwerk-2.6.12
Parsing documentation for thor-1.2.2
Installing ri documentation for thor-1.2.2
Parsing documentation for method_source-1.0.0
Installing ri documentation for method_source-1.0.0
Parsing documentation for concurrent-ruby-1.2.2
Installing ri documentation for concurrent-ruby-1.2.2
Parsing documentation for tzinfo-2.0.6
Installing ri documentation for tzinfo-2.0.6
Parsing documentation for minitest-5.20.0
Installing ri documentation for minitest-5.20.0
Parsing documentation for i18n-1.14.1
Installing ri documentation for i18n-1.14.1
Parsing documentation for activesupport-7.0.8
Installing ri documentation for activesupport-7.0.8
Parsing documentation for nokogiri-1.15.4-x86_64-linux
Installing ri documentation for nokogiri-1.15.4-x86_64-linux
Parsing documentation for crass-1.0.6
Installing ri documentation for crass-1.0.6
Parsing documentation for loofah-2.21.3
Installing ri documentation for loofah-2.21.3
Parsing documentation for rails-html-sanitizer-1.6.0
Installing ri documentation for rails-html-sanitizer-1.6.0
Parsing documentation for rails-dom-testing-2.2.0
Installing ri documentation for rails-dom-testing-2.2.0
Parsing documentation for rack-2.2.8
Installing ri documentation for rack-2.2.8
Parsing documentation for rack-test-2.1.0
Installing ri documentation for rack-test-2.1.0
Parsing documentation for erubi-1.12.0
Installing ri documentation for erubi-1.12.0
Parsing documentation for builder-3.2.4
Installing ri documentation for builder-3.2.4
Parsing documentation for actionview-7.0.8
Installing ri documentation for actionview-7.0.8
Parsing documentation for actionpack-7.0.8
Installing ri documentation for actionpack-7.0.8
Parsing documentation for railties-7.0.8
Installing ri documentation for railties-7.0.8
Parsing documentation for mini_mime-1.1.5
Installing ri documentation for mini_mime-1.1.5
Parsing documentation for marcel-1.0.2
Installing ri documentation for marcel-1.0.2
Parsing documentation for activemodel-7.0.8
Installing ri documentation for activemodel-7.0.8
Parsing documentation for activerecord-7.0.8
Installing ri documentation for activerecord-7.0.8
Parsing documentation for globalid-1.2.1
Installing ri documentation for globalid-1.2.1
Parsing documentation for activejob-7.0.8
```

```

Installing ri documentation for activejob-7.0.8
Parsing documentation for activestorage-7.0.8
Installing ri documentation for activestorage-7.0.8
Parsing documentation for actiontext-7.0.8
Installing ri documentation for actiontext-7.0.8
Parsing documentation for mail-2.8.1
Installing ri documentation for mail-2.8.1
Parsing documentation for actionmailer-7.0.8
Installing ri documentation for actionmailer-7.0.8
Parsing documentation for actionmailbox-7.0.8
Installing ri documentation for actionmailbox-7.0.8
Parsing documentation for websocket-extensions-0.1.5
Installing ri documentation for websocket-extensions-0.1.5
Parsing documentation for websocket-driver-0.7.6
Installing ri documentation for websocket-driver-0.7.6
Parsing documentation for nio4r-2.5.9
Installing ri documentation for nio4r-2.5.9
Parsing documentation for actioncable-7.0.8
Installing ri documentation for actioncable-7.0.8
Parsing documentation for rails-7.0.8
Installing ri documentation for rails-7.0.8
Done installing documentation for zeitwerk, thor, method_source, concurrent-
ruby, tzinfo, minitest, i18n, activesupport, nokogiri, crass, loofah,
rails-html-sanitizer, rails-dom-testing, rack, rack-test, erubi, builder
, actionview, actionpack, railties, mini_mime, marcel, activemodel,
activerecord, globalid, activejob, activestorage, actiontext, mail,
actionmailer, actionmailbox, websocket-extensions, websocket-driver,
nio4r, actioncable, rails after 32 seconds
36 gems installed

```

A Rails rendszerünk konfigurációja a következő témánk. Kétféle web-szervert használunk, a fejlesztéshez a Puma-t, míg az éles rendszerhez az Apache2-t. Adatbáziskezelőből szintén kétfélét nézünk meg, a beágyazott SQLite3-at és a kliens-szerver alapú MySQL/MariaDB-t. A következőkben ezek illesztéséhez szükséges adapterek fordítását lehetővé tevő C++ és Ruby függvénykönyvtárakat telepítjük.

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

```

root@debian:~# gem install passenger
Fetching passenger-6.0.18.gem
Building native extensions. This could take a while...
Successfully installed passenger-6.0.18
Parsing documentation for passenger-6.0.18
Installing ri documentation for passenger-6.0.18
Done installing documentation for passenger after 26 seconds
1 gem installed

```

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@debian:~# apt-get install libmariadb-dev
Reading package lists... Done
Building dependency tree... Done

```

```

Reading state information... Done
The following additional packages will be installed:
  zlib1g-dev
The following NEW packages will be installed:
  libmariadb-dev zlib1g-dev
0 upgraded, 2 newly installed, 0 to remove and 16 not upgraded.
Need to get 0 B/1,180 kB of archives.
After this operation, 2,302 kB of additional disk space will be used.
Do you want to continue? [Y/n]
Selecting previously unselected package zlib1g-dev:amd64.
(Reading database ... 243557 files and directories currently installed.)
Preparing to unpack .../zlib1g-dev_1%3a1.2.13.dfsg-1_amd64.deb ...
Unpacking zlib1g-dev:amd64 (1:1.2.13.dfsg-1) ...
Selecting previously unselected package libmariadb-dev.
Preparing to unpack .../libmariadb-dev_1%3a10.11.3-1_amd64.deb ...
Unpacking libmariadb-dev (1:10.11.3-1) ...
Setting up zlib1g-dev:amd64 (1:1.2.13.dfsg-1) ...
Setting up libmariadb-dev (1:10.11.3-1) ...
Processing triggers for man-db (2.11.2-2) ...

```

Ezután telepíthetjük magukat az adatbáziskezelő-adaptereket, a `sqlite3`-t és a `mysql2`-t. A PostgreSQL adapterét `pg`-nek hívják, a gyakorlaton azt nem fogjuk használni.

```

root@debian:~# gem install sqlite3
Fetching sqlite3-1.6.6-x86_64-linux.gem
Successfully installed sqlite3-1.6.6-x86_64-linux
Parsing documentation for sqlite3-1.6.6-x86_64-linux
Installing ri documentation for sqlite3-1.6.6-x86_64-linux
Done installing documentation for sqlite3 after 1 seconds
1 gem installed

```

```

root@debian:~# gem install mysql2
Building native extensions. This could take a while...
Successfully installed mysql2-0.5.5
Parsing documentation for mysql2-0.5.5
Installing ri documentation for mysql2-0.5.5
Done installing documentation for mysql2 after 0 seconds
1 gem installed

```

Az alap Rails rendszerünk összeállt, ha a Rails 6-os verziója előtti változatot használunk, akkor szükségünk van további gemek telepítésére. Ilyen a `mini_racer`, ami egy szerver oldali JavaScript interpreter, vagy a `turbolinks` API nélkül a Rails már nem hajlandó elindulni, noha az opcionális. Rails 6 verziójában a Rails alkalmazásba intergált a Node.js, amely a 7-es verziótól kezdve ismét opcionális.

A Rails-en belüli csomagkezeléshez szükségünk van a `bundler` gemre, amelyet a Rails majd automatikusan telepít.

A Rails telepítése során a `racc` függvénykönyvtárnak szüksége lesz a `yaml.h` C header fájlra, amely a `libyaml-dev` csomagban található.

```

root@debian:~# apt-get install libyaml-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:

```

```

libyaml-doc
The following NEW packages will be installed:
libyaml-dev
0 upgraded, 1 newly installed, 0 to remove and 16 not upgraded.
Need to get 0 B/64.5 kB of archives.
After this operation, 262 kB of additional disk space will be used.
Selecting previously unselected package libyaml-dev:amd64.
(Reading database ... 243753 files and directories currently installed.)
Preparing to unpack .../libyaml-dev_0.2.5-1_amd64.deb ...
Unpacking libyaml-dev:amd64 (0.2.5-1) ...
Setting up libyaml-dev:amd64 (0.2.5-1) ...

```

2. Rails alkalmazás létrehozása

A Rails használatához a rails szkriptet használjuk immáron nem rendszer-gazdaként, hanem egyszerű felhasználóként. A parancs Debian/Ubuntu Linux és 3.1-es ruby esetén a /usr/local/bin/ könyvtárba került.

A -d kapcsoló különös jelentőséggel bír számunkra, ezzel adhatjuk meg a használni kívánt adatbáziskezelő típusát. A gyakorlatok keretében ezek közül a sqlite-ot mint alapértelmezett adatbáziskezelőt és a mysql-t fogjuk használni. A -B kapcsoló mellőzi a Rails alkalmazásunk Ruby függőségeinek feloldását, amelyet azonban később pótolnunk kell. A -J kapcsolóval pedig későbbre halasztjuk a Javascript csomagkezelő kiválasztását.

```

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

Options:
  [--skip-namespace], [--no-skip-namespace]           # Skip
                                                         namespace (affects only isolated engines)
  [--skip-collision-check], [--no-skip-collision-check] # Skip
                                                         collision check
-r, --ruby=PATH                                       # Path to
                                                         the Ruby binary of your choice
                                                         # Default
                                                         : /
                                                         usr/
                                                         bin/
                                                         ruby
-m, --template=TEMPLATE                               # Path to
                                                         some application template (can be a filesystem path or URL)
-d, --database=DATABASE                               #
                                                         Preconfigure for selected database (options: mysql/postgresql/sqlite3/
                                                         oracle/sqlserver/jdbcmysql/jdbcsqlite3/jdbcpostgresql/jdbc)
                                                         # Default
                                                         :
                                                         sqlite3
-G, --skip-git, [--no-skip-git]                       # Skip
                                                         gitignore file
  [--skip-keeps], [--no-skip-keeps]                   # Skip
                                                         source control .keep files

```



```

-M,      [--skip-action-mailer], [--no-skip-action-mailer]      # Skip
      Action Mailer files
      [--skip-action-mailbox], [--no-skip-action-mailbox]      # Skip
      Action Mailbox gem
      [--skip-action-text], [--no-skip-action-text]            # Skip
      Action Text gem
-O,      [--skip-active-record], [--no-skip-active-record]      # Skip
      Active Record files
      [--skip-active-job], [--no-skip-active-job]              # Skip
      Active Job
      [--skip-active-storage], [--no-skip-active-storage]      # Skip
      Active Storage files
-C,      [--skip-action-cable], [--no-skip-action-cable]        # Skip
      Action Cable files
-A,      [--skip-asset-pipeline], [--no-skip-asset-pipeline]    #
      Indicates when to generate skip asset pipeline
-a,      [--asset-pipeline=ASSET_PIPELINE]                      # Choose
      your asset pipeline [options: sprockets (default), propshaft]
                                                    # Default
                                                    :
                                                    sprockets
-J,      [--skip-javascript], [--no-skip-javascript]            # Skip
      JavaScript files
      [--skip-hotwire], [--no-skip-hotwire]                    # Skip
      Hotwire integration
      [--skip-jbuilder], [--no-skip-jbuilder]                  # Skip
      jbuilder gem
-T,      [--skip-test], [--no-skip-test]                        # Skip
      test files
      [--skip-system-test], [--no-skip-system-test]            # Skip
      system test files
      [--skip-bootsnap], [--no-skip-bootsnap]                  # Skip
      bootsnap gem
      [--dev], [--no-dev]                                      # Set up
      the application with Gemfile pointing to your Rails checkout
      [--edge], [--no-edge]                                    # Set up
      the application with Gemfile pointing to Rails repository
--master, [--main], [--no-main]                                 # Set up
      the application with Gemfile pointing to Rails repository main branch
      [--rc=RC]                                                # Path to
      file containing extra configuration options for rails
      command
      [--no-rc], [--no-no-rc]                                  # Skip
      loading of extra configuration options from .railsrc file
      [--api], [--no-api]                                      #
      Preconfigure smaller stack for API only apps
      [--minimal], [--no-minimal]                             #
      Preconfigure a minimal rails app
-j,      [--javascript=JAVASCRIPT]                              # Choose
      JavaScript approach [options: importmap (default), webpack, esbuild,
      rollup]
                                                    # Default
                                                    :
                                                    importmap
-c,      [--css=CSS]                                             # Choose
      CSS processor [options: tailwind, bootstrap, bulma, postcss, sass]
      check https://github.com/rails/cssbundling-rails
-B,      [--skip-bundle], [--no-skip-bundle]                    # Don't
      run bundle install

```

```

Runtime options:
  -f, [--force] # Overwrite files that already exist
  -p, [--pretend], [--no-pretend] # Run but do not make any changes
  -q, [--quiet], [--no-quiet] # Suppress status output
  -s, [--skip], [--no-skip] # Skip files that already exist

Rails options:
  -h, [--help], [--no-help] # Show this help message and quit
  -v, [--version], [--no-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,
  or in $XDG_CONFIG_HOME/rails/railsrc if XDG_CONFIG_HOME is set.

  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.

```

A Rails keretrendszerben a rails parancs az az univerzális eszköz mellyel többek között új komponenseket hozhatunk létre és elindíthatjuk a beépített webszervert. Egy új Rails alkalmazást a rails parancsnak new opciót megadva hozhatunk létre, a második argumentum az alkalmazás neve. A parancsot tetszőleges felhasználóként 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 lefutna a Rails keretrendszer egy másik parancsa, a bundle, amely az install opció hatására összeszedi a gemek közül azokat, amelyekre az alkalmazásunknak szüksége lesz, azonban ezt a -B kapcsolóval letiltjuk, mert testre akarjuk szabni a keretrendszerünket. A Node.js telepítését a -J kapcsolóval tilthatjuk le.

```

kovacs@debian:~> rails new gyakorlat -J -B
  create
  create  README.md
  create  Rakefile
  create  .ruby-version
  create  config.ru
  create  .gitignore
  create  .gitattributes
  create  Gemfile
  run    git init from "."
hint: Using 'master' as the name for the initial branch. This default branch
name
hint: is subject to change. To configure the initial branch name to use in
all
hint: of your new repositories, which will suppress this warning, call:

```

```

hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command
:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/kovacs/gyakorlat/.git/
create  app
create  app/assets/config/manifest.js
create  app/assets/stylesheets/application.css
create  app/channels/application_cable/channel.rb
create  app/channels/application_cable/connection.rb
create  app/controllers/application_controller.rb
create  app/helpers/application_helper.rb
create  app/jobs/application_job.rb
create  app/mailers/application_mailer.rb
create  app/models/application_record.rb
create  app/views/layouts/application.html.erb
create  app/views/layouts/mailer.html.erb
create  app/views/layouts/mailer.text.erb
create  app/assets/images
create  app/assets/images/.keep
create  app/controllers/concerns/.keep
create  app/models/concerns/.keep
create  bin
create  bin/rails
create  bin/rake
create  bin/setup
create  config
create  config/routes.rb
create  config/application.rb
create  config/environment.rb
create  config/cable.yml
create  config/puma.rb
create  config/storage.yml
create  config/environments
create  config/environments/development.rb
create  config/environments/production.rb
create  config/environments/test.rb
create  config/initializers
create  config/initializers/assets.rb
create  config/initializers/content_security_policy.rb
create  config/initializers/cors.rb
create  config/initializers/filter_parameter_logging.rb
create  config/initializers/inflections.rb
create  config/initializers/new_framework_defaults_7_0.rb
create  config/initializers/permissions_policy.rb
create  config/locales
create  config/locales/en.yml
create  config/master.key
append  .gitignore
create  config/boot.rb
create  config/database.yml
create  db
create  db/seeds.rb
create  lib
create  lib/tasks
create  lib/tasks/.keep
create  lib/assets
create  lib/assets/.keep

```

```
create log
create log/.keep
create public
create public/404.html
create public/422.html
create public/500.html
create public/apple-touch-icon-precomposed.png
create public/apple-touch-icon.png
create public/favicon.ico
create public/robots.txt
create tmp
create tmp/.keep
create tmp/pids
create tmp/pids/.keep
create tmp/cache
create tmp/cache/assets
create vendor
create vendor/.keep
create test/fixtures/files
create test/fixtures/files/.keep
create test/controllers
create test/controllers/.keep
create test/mailers
create test/mailers/.keep
create test/models
create test/models/.keep
create test/helpers
create test/helpers/.keep
create test/integration
create test/integration/.keep
create test/channels/application_cable/connection_test.rb
create test/test_helper.rb
create test/system
create test/system/.keep
create test/application_system_test_case.rb
create storage
create storage/.keep
create tmp/storage
create tmp/storage/.keep
remove config/initializers/cors.rb
remove config/initializers/new_framework_defaults_7_0.rb
```

A kimenet végén azt olvashatjuk, hogy futtatnunk kell a `bundle install`, illetve a `rails webpacker:install` parancsokat a telepítés befejezéséhez, de előtte nézzük meg a létrehozott fájlokat!

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 `bin` könyvtár elérhetővé teszi számunkra a `rails`, a `rake` és a `bundle` parancsokat, amelyeket a Rails alkalmazásunk menedzsmentjét fogjuk megvalósítani. 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 szerIALIZED formában magát az adatbázis tartalmazza. 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.

Rails 6 esetén `Node.js` a `node_modules` könyvtárba kerül, függőségeiket a `package.json` fájl tartalmazza, és a `bin/yarn` paranccsal kezelhetjük a telepített JavaScript API-kat.

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 érhetőek el.

Az első dolgunk a `Gemfile` módosítása. Először kikommentezzük az első sort, hogy a Rails a Ruby függvénykönyvtárakat a helyi, frissen telepített helyen keresse, és ne a megadott távoli szerveren. Ez azért szükséges, hogy a rendszer egy rögzített verziójú keretrendszer mellett tudhassuk fejleszteni, és az ne frissüljön a `Gemfile` minden módosításakor. Egy esetleges frissítés fejlesztés közben elronthatja a teljes addigi munkánkat! A `Gemfile` minden további módosítása után futtatnunk kell majd a `bundle install` vagy `bundle update` parancsot.

A konzolon kiadott `bundle install` paranccsal 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. A `bundle` parancshoz a `--path` kapcsolót és egy fájlrendszeri útvonalat hozzáfűzve elérhetjük, hogy garantáltan mindig a saját Ruby környezetünkkel dolgozhassunk, és ne ütközzünk az időközben bekövetkező rendszerfrissítések okozta inkompatibilitás áldozatául. Ha a Ruby csomagokat minden felhasználó számára elérhetővé akarjuk tenni, meg kell adnunk az adminisztrátor jelszavát.⁴

Az SQLite3 adatbáziskezelővel konfigurált Rails alkalmazás adatbázis kapcsolat-leíró állományának vizsgálata után újrageneráljuk a projektet immáron MariaDB adatbáziskezelővel. A különbség a `config/database.yml` és a `Gemfile` fájlokban jelentkezik mindössze.

```
kovacsg@debian:~> rails new gyakorlat -d mysql -J -B
```

Telepítsük a függőségeket. Mivel a `ruby-t` rendszergazdaként operációs rendszer szinten telepítettük, a Rails alkalmazást pedig felhasználóként a felhasználói doménben, ahhoz, hogy a még hiányzó Ruby API-kat telepíteni tudjunk, a parancs elkéri a jelszavunkat a `sudo` parancshoz.

⁴A Rails telepítése után nem minden csomag áll rendelkezésre a `bundle` számára, az további csomagok telepítésére kérni fogja a rendszergazdai jelszót. A gyakorlaton csalást követtünk el, az összes további csomag már előtelepített állapotban volt.

```
kovacs@debian:~/gyakorlat> bundle install
Fetching gem metadata from https://rubygems.org/.....
Resolving dependencies.....
Using rake 13.0.6
Following files may not be writable, so sudo is needed:
  /usr/local/bin
  /var/lib/gems/3.1.0
  /var/lib/gems/3.1.0/build_info
  /var/lib/gems/3.1.0/cache
  /var/lib/gems/3.1.0/doc
  /var/lib/gems/3.1.0/extensions
  /var/lib/gems/3.1.0/gems
  /var/lib/gems/3.1.0/plugins
  /var/lib/gems/3.1.0/specifications
Using concurrent-ruby 1.2.2
Using il8n 1.14.1
Using minitest 5.20.0
Using tzinfo 2.0.6
Using activesupport 7.0.8
Using builder 3.2.4
Using erubi 1.12.0
Using racc 1.7.1
Using nokogiri 1.15.4 (x86_64-linux)
Using rails-dom-testing 2.2.0
Using crass 1.0.6
Using loofah 2.21.3
Using rails-html-sanitizer 1.6.0
Using actionview 7.0.8
Using rack 2.2.8
Using rack-test 2.1.0
Using actionpack 7.0.8
Using nio4r 2.5.9
Using websocket-extensions 0.1.5
Using websocket-driver 0.7.6
Using actioncable 7.0.8
Using globalid 1.2.1
Using activejob 7.0.8
Using activemodel 7.0.8
Using activerecord 7.0.8
Using marcel 1.0.2
Using mini_mime 1.1.5
Using activestorage 7.0.8
Using date 3.3.3
Using timeout 0.4.0
Using net-protocol 0.2.1
Using net-imap 0.3.7
Using net-pop 0.1.2
Using net-smtp 0.4.0
Using mail 2.8.1
Using actionmailbox 7.0.8
Using actionmailer 7.0.8
Using actiontext 7.0.8
Using public_suffix 5.0.3
Using addressable 2.8.5
Using bindindex 0.8.1
Using msgpack 1.7.2
Using bootsnap 1.16.0
Using bundler 2.3.7
Using matrix 0.4.2
Using regexp_parser 2.8.1
Using xpath 3.2.0
```

```

Using capybara 3.39.2
Using stringio 3.0.8
Using psych 5.1.0
Using rdoc 6.5.0
Using io-console 0.6.0
Fetching reline 0.3.8

Your user account isn't allowed to install to the system RubyGems.
You can cancel this installation and run:

bundle config set --local path 'vendor/bundle'
bundle install

to install the gems into ./vendor/bundle/, or you can enter your password
and install the bundled gems to RubyGems using sudo.

Password:
Sorry, try again.

Your user account isn't allowed to install to the system RubyGems.
You can cancel this installation and run:

bundle config set --local path 'vendor/bundle'
bundle install

to install the gems into ./vendor/bundle/, or you can enter your password
and install the bundled gems to RubyGems using sudo.

Password:

Your user account isn't allowed to install to the system RubyGems.
You can cancel this installation and run:

bundle config set --local path 'vendor/bundle'
bundle install

to install the gems into ./vendor/bundle/, or you can enter your password
and install the bundled gems to RubyGems using sudo.

Password:
Installing reline 0.3.8
Fetching irb 1.8.1
Installing irb 1.8.1
Fetching debug 1.8.0
Installing debug 1.8.0 with native extensions
Fetching jbuilder 2.11.5
Installing jbuilder 2.11.5
Using method_source 1.0.0
Using mysql2 0.5.5
Fetching puma 5.6.7
Installing puma 5.6.7 with native extensions
Using thor 1.2.2
Using zeitwerk 2.6.12
Using railties 7.0.8
Using rails 7.0.8
Fetching rexml 3.2.6
Installing rexml 3.2.6
Fetching rubyzip 2.3.2
Installing rubyzip 2.3.2

```

```
Fetching websocket 1.2.10
Installing websocket 1.2.10
Fetching selenium-webdriver 4.13.1
Installing selenium-webdriver 4.13.1
Fetching sprockets 4.2.1
Installing sprockets 4.2.1
Fetching sprockets-rails 3.4.2
Installing sprockets-rails 3.4.2
Fetching web-console 4.2.1
Installing web-console 4.2.1
Bundle complete! 11 Gemfile dependencies, 71 gems now installed.
Use 'bundle info [gemname]' to see where a bundled gem is installed.
Post-install message from rubyzip:
RubyZip 3.0 is coming!
*****

The public API of some Rubyzip classes has been modernized to use named
parameters for optional arguments. Please check your usage of the
following classes:
** Zip::File **
** Zip::Entry **
** Zip::InputStream **
** Zip::OutputStream **

Please ensure that your Gemfiles and .gemspecs are suitably restrictive
to avoid an unexpected breakage when 3.0 is released (e.g. ~> 2.3.0).
See https://github.com/rubyzip/rubyzip for details. The Changelog also
lists other enhancements and bugfixes that have been implemented since
version 2.3.0.
```

A Rails rendszerünk használatához minden függőséget feloldottunk, ezután az alábbi Ruby függvénykönyvtárak kerültek telepítésre. Az `action` és az `active` kezdetű függvénykönyvtárak adják a Rails keretrendszer magját.

```
kovacs@debian:~/gyakorlat> gem list

*** LOCAL GEMS ***

abbrev (default: 0.1.0)
actioncable (7.0.8)
actionmailbox (7.0.8)
actionmailer (7.0.8)
actionpack (7.0.8)
actiontext (7.0.8)
actionview (7.0.8)
activejob (7.0.8)
activemodel (7.0.8)
activerecord (7.0.8)
activestorage (7.0.8)
activesupport (7.0.8)
addressable (2.8.5)
base64 (default: 0.1.1)
benchmark (default: 0.2.0)
bigdecimal (default: 3.1.1)
bindex (0.8.1)
bootsnap (1.16.0)
builder (3.2.4)
bundler (default: 2.3.7)
capybara (3.39.2)
cgi (default: 0.3.5)
concurrent-ruby (1.2.2)
```



```
crass (1.0.6)
csv (default: 3.2.2)
date (3.3.3, default: 3.2.2)
debug (1.8.0, 1.4.0)
delegate (default: 0.2.0)
did_you_mean (default: 1.6.1)
digest (default: 3.1.0)
drb (default: 2.1.0)
english (default: 0.7.1)
erb (default: 2.2.3)
error_highlight (default: 0.3.0)
erubi (1.12.0)
etc (default: 1.3.0)
fcntl (default: 1.0.1)
fiddle (default: 1.1.0)
fileutils (default: 1.6.0)
find (default: 0.1.1)
forwardable (default: 1.3.2)
getoptlong (default: 0.1.1)
globalid (1.2.1)
i18n (1.14.1)
io-console (0.6.0, default: 0.5.11)
io-nonblock (default: 0.1.0)
io-wait (default: 0.2.1)
ipaddr (default: 1.2.4)
irb (1.8.1, default: 1.4.1)
jbuilder (2.11.5)
json (default: 2.6.1)
logger (default: 1.5.0)
loofah (2.21.3)
mail (2.8.1)
marcel (1.0.2)
matrix (0.4.2)
method_source (1.0.0)
mini_mime (1.1.5)
minitest (5.20.0, 5.15.0)
msgpack (1.7.2)
mutex_m (default: 0.1.1)
mysql2 (0.5.5)
net-ftp (0.1.3)
net-http (default: 0.2.0)
net-imap (0.3.7, 0.2.3)
net-pop (0.1.2, 0.1.1)
net-protocol (0.2.1, default: 0.1.2)
net-smtp (0.4.0, 0.3.1)
net-telnet (0.2.0)
nio4r (2.5.9)
nkf (default: 0.1.1)
nokogiri (1.15.4 x86_64-linux)
observer (default: 0.1.1)
open-uri (default: 0.2.0)
open3 (default: 0.1.1)
openssl (default: 3.0.1)
optparse (default: 0.2.0)
ostruct (default: 0.5.2)
passenger (6.0.18)
pathname (default: 0.2.0)
power_assert (2.0.1)
pp (default: 0.3.0)
prettyprint (default: 0.1.1)
prime (0.1.2)
pstore (default: 0.1.1)
```

```
psych (5.1.0, default: 4.0.3)
public_suffix (5.0.3)
puma (5.6.7)
racc (1.7.1, default: 1.6.0)
rack (3.0.8, 2.2.8)
rack-test (2.1.0)
rails (7.0.8)
rails-dom-testing (2.2.0)
rails-html-sanitizer (1.6.0)
railties (7.0.8)
rake (13.0.6)
rbs (2.1.0)
rdoc (6.5.0, default: 6.4.0)
readline (default: 0.0.3)
readline-ext (default: 0.1.4)
regexp_parser (2.8.1)
reline (0.3.8, default: 0.3.0)
resolv (default: 0.2.1)
resolv-replace (default: 0.1.0)
rexml (3.2.6, 3.2.5)
rinda (default: 0.1.1)
rss (0.2.9)
ruby2_keywords (default: 0.0.5)
rubygems-update (3.3.15)
rubyzip (2.3.2)
sdbm (1.0.0)
securerandom (default: 0.1.1)
selenium-webdriver (4.13.1)
set (default: 1.0.2)
shellwords (default: 0.1.0)
singleton (default: 0.1.1)
sprockets (4.2.1)
sprockets-rails (3.4.2)
sqlite3 (1.6.6 x86_64-linux)
stringio (3.0.8, default: 3.0.1)
strscan (default: 3.0.1)
syslog (default: 0.1.0)
tempfile (default: 0.1.2)
test-unit (3.5.3)
thor (1.2.2)
time (default: 0.2.0)
timeout (0.4.0, default: 0.2.0)
tmpdir (default: 0.1.2)
tsort (default: 0.1.0)
typeprof (0.21.2)
tzinfo (2.0.6)
un (default: 0.2.0)
uri (default: 0.11.0)
weakref (default: 0.1.1)
web-console (4.2.1)
webrick (1.8.1)
websocket (1.2.10)
websocket-driver (0.7.6)
websocket-extensions (0.1.5)
xmlrpc (0.3.2)
xpath (3.2.0)
yaml (default: 0.2.0)
zeitwerk (2.6.12)
zlib (default: 2.1.1)
```

A rails parancsot az újonnan létrehozott Rails alkalmazásunk valamely

alkönyvtárából kiadva más üzenetet látunk, mint az előbb, az alkalmazás létrehozása előtt. Az új alkalmazást létrehozó **new** mellett jelen van több más mellett a kódgenerálásra alkalmas **generate** opció, a beágyazott webservert indító **server** opció, a Ruby, illetve az adatbáziskonzolt indító **console**, illetve **dbconsole** opciók.

```
kovacs@debian:~/gyakorlat> rails
The most common rails commands are:
generate      Generate new code (short-cut alias: "g")
console       Start the Rails console (short-cut alias: "c")
server        Start the Rails server (short-cut alias: "s")
test          Run tests except system tests (short-cut alias: "t")
test:system   Run system tests
dbconsole     Start a console for the database specified in config/database.
              yml
              (short-cut alias: "db")
new           Create a new Rails application. "rails_new_my_app" creates a
              new application called MyApp in "./my_app"
plugin new    Create a new Rails railtie or engine

All commands can be run with -h (or —help) for more information.
In addition to those commands, there are:

about
action_mailbox:ingress:exim
action_mailbox:ingress:postfix
action_mailbox:ingress:qmail
action_mailbox:install
action_mailbox:install:migrations
action_text:install
action_text:install:migrations
active_storage:install
app:template
app:update
assets:clean[keep]
assets:clobber
assets:environment
assets:precompile
cache_digests:dependencies
cache_digests:nested_dependencies
credentials:diff
credentials:edit
credentials:show
db:create
db:drop
db:encryption:init
db:environment:set
db:fixtures:load
db:migrate
db:migrate:down
db:migrate:redo
db:migrate:status
db:migrate:up
db:prepare
db:reset
db:rollback
db:schema:cache:clear
db:schema:cache:dump
db:schema:dump
db:schema:load
```

```
db:seed
db:seed:replant
db:setup
db:system:change
db:version
destroy
dev:cache
encrypted:edit
encrypted:show
initializers
log:clear
middleware
notes
restart
routes
runner
secret
secrets:edit
secrets:setup
secrets:show
stats
test:all
test:db
time:zones[country_or_offset]
tmp:clear
tmp:create
version
yarn:install
zeitwerk:check
```

A `rails` parancs gyakran használt argumentuma a `console` vagy röviden `c`, amivel az első gyakorlatról megismert interaktív Ruby értelmezőt indíthatunk, melyben a Rails alkalmazásunk környezeti beállításai inicializálásra kerültek.

```
kovacs@debian:~/gyakorlat$ rails c
Loading development environment (Rails 7.0.8)
irb(main):001> Rails.application
=> #<Gyakorlat::Application>
irb(main):002>
```

3. Web- és adatbázisszerver kapcsolat beállítása

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

Az alapértelmezett webservert, a Puma-t a `rails server` vagy röviden `s` opciójával indíthatjuk el a Rails alkalmazásunk tetszőleges könyvtárából. Ha a szerveret háttérszolgáltatásként kívánjuk futtatni, akkor a `-d` kapcsolót is hozzá kell fűznünk a parancssorhoz. Ezután a szerver elérhető a `http://localhost:3000` webcímen. Ha a `-b` kapcsoló után megadjuk a virtuális gép IP címét, akkor a weboldalhoz a virtuális gépen kívülről is, például a

gazdagépről is hozzáférhetünk. Ha megnyitjuk az oldalt egy böngészőben, akkor annak eseményei megjelennek a konzolon, és bekerülnek a naplófájlba is. Itt azt látjuk, hogy HTTP GET művelet fért hozzá a "/" erőforráshoz, és, hogy a Rails ennek hatására milyen műveleteket végzett el. A kimeneten láthatjuk, hogy a webszerver a fejlesztői környezet konfigurációját használja.

```
kovacs@debian:~/gyakorlat> rails s
=> Booting Puma
=> Rails 7.0.8 application starting in development
=> Run 'bin/rails server --help' for more startup options
Puma starting in single mode...
* Puma version: 5.6.7 (ruby 3.1.2-p20) ("Birdie's_Version")
* Min threads: 5
* Max threads: 5
* Environment: development
* PID: 162153
* Listening on http://127.0.0.1:3000
* Listening on http://[::]:3000
Use Ctrl-C to stop
kovacs@debian:~/gyakorlat$ rails s -b 10.211.55.3
=> Booting Puma
=> Rails 7.0.8 application starting in development
=> Run 'bin/rails server --help' for more startup options
Puma starting in single mode...
* Puma version: 5.6.7 (ruby 3.1.2-p20) ("Birdie's_Version")
* Min threads: 5
* Max threads: 5
* Environment: development
* PID: 163299
* Listening on http://10.211.55.3:3000
Use Ctrl-C to stop
```

A Rails a már említett Passenger plugin segítségével illeszthető Apache2 webszerverhez. Ehhez az alábbi kódrészletben mutatott parancsot kell kiadnunk. A parancs lefordítja, majd telepíti az Apache webszerver Rails (és egyéb további) 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. Az alábbi naplórészlet a Passenger plugin telepítését mutatja be, a C források fordítása nélkül.

```
root@debian:~# passenger-install-apache2-module
Welcome to the Phusion Passenger Apache 2 module installer, v6.0.18.

This installer will guide you through the entire installation process. It
shouldn't take more than 3 minutes in total.

Here's what you can expect from the installation process:

1. The Apache 2 module will be installed for you.
2. You'll learn how to configure Apache.
3. You'll learn how to deploy a Ruby on Rails application.
```

Don't worry if anything goes wrong. This installer will advise you on how to solve any problems.

Press Enter to continue, or Ctrl-C to abort.

Which languages are you interested in?

Use <space> to select.

If the menu doesn't display correctly, press '!'.

```
> x  Ruby
    _ Python
    _ Node.js
    _ Meteor
```

Checking for required software...

```
* Checking for C compiler...
  Found: yes
  Location: /usr/bin/cc
* Checking for C++ compiler...
  Found: yes
  Location: /usr/bin/c++
* Checking for Curl development headers with SSL support...
  Found: yes
  curl-config location: /usr/bin/curl-config
  Header location: /usr/include/x86_64-linux-gnu/curl/curl.h
  Version: libcurl 7.88.1
  Usable: yes
  Supports SSL: yes
* Checking for Zlib development headers...
  Found: yes
  Location: /usr/include/zlib.h
* Checking for Apache 2...
  Found: yes
  Location of httpd: /usr/sbin/apache2
  Apache version: 2.4.57
* Checking for Rake (associated with /usr/bin/ruby3.1)...
  Found: yes
  Location: /usr/bin/ruby3.1 /usr/bin/rake
* Checking for OpenSSL support for Ruby...
  Found: yes
* Checking for RubyGems...
  Found: yes
* Checking for Ruby development headers...
  Found: yes
  Location: /usr/include/ruby-3.1.0/ruby.h
* Checking for rack...
  Found: yes
* Checking for OpenSSL development headers...
  Found: yes
  Location: /usr/include/openssl/ssl.h
* Checking for Apache 2 development headers...
  Found: yes
  Location of apxs2: /usr/bin/apxs2
* Checking for Apache Portable Runtime (APR) development headers...
```

```
Found: yes
Location: /usr/bin/apr-1-config
Version: 1.7.2
* Checking for Apache Portable Runtime Utility (APU) development headers...
Found: yes
Location: /usr/bin/apu-1-config
Version: 1.6.3

-----

Checking whether there are multiple Apache installations...
Only a single installation detected. This is good.

-----

Compiling and installing Apache 2 module...

-----

Almost there!

Please edit your Apache configuration file, and add these lines:

    LoadModule passenger_module /var/lib/gems/3.1.0/gems/passenger-6.0.18/
      buildout/apache2/mod_passenger.so
    <IfModule mod_passenger.c>
      PassengerRoot /var/lib/gems/3.1.0/gems/passenger-6.0.18
      PassengerDefaultRuby /usr/bin/ruby3.1
    </IfModule>

After you restart Apache, you are ready to deploy any number of web
applications on Apache, with a minimum amount of configuration!

Press ENTER when you are done editing.

-----

Validating installation...

* Checking whether this Passenger install is in PATH...
* Checking whether there are no other Passenger installations...
* Checking whether Apache is installed...
* Checking whether the Passenger module is correctly configured in Apache
  ...

Everything looks good. :-)
```

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 Apache2 modul elérhetővé tételére az Apache2 konfigurációs könyvtárban létre kell hoznunk a modul beállításait. Szerencsére a telepítő parancs a helyes beállításokat kírja a konzolra. A fájl `/etc/apache2/mods-available/`

`rails.load` néven hozzuk létre, a tartalma az alábbi konfigurációrészlethez hasonló. A modult a `a2enmod rails` paranccsal, majd a webszerver újraindításával tehetjük aktívvá.

```
LoadModule passenger_module /var/lib/gems/3.1.0/gems/passenger-6.0.18/
  buildout/apache2/mod_passenger.so
<IfModule mod_passenger.c>
  PassengerRoot /var/lib/gems/3.1.0/gems/passenger-6.0.18
  PassengerDefaultRuby /usr/bin/ruby3.1
</IfModule>
```

Az Apache2 virtuális hoszt beállításait szintén konfigurációs állományok között kell elhelyeznünk például `/etc/apache2/sites-available/gyakorlat.conf` néven, majd az `a2ensite gyakorlat.conf` paranccsal konzolon engedélyezzük a gyakorlat oldalt⁵. Ennek tartalma a telepítő parancs alapján a következő lehet:

```
RackEnv development

#<VirtualHost *:80>
<VirtualHost www.gyakorlat.com:80>

  ServerName www.gyakorlat.com
  ServerAdmin admin@gyakorlat.com
  DocumentRoot /home/kovacs/gyakorlat/public
  ServerSignature On

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

  <Directory />
    Options FollowSymLinks
    AllowOverride None
  </Directory>
  <Directory /home/kovacs/gyakorlat/public/>
    Require all granted
    Options Indexes FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all
    Options -MultiViews
  </Directory>

</VirtualHost>
```

Mivel módosítottuk az Apache2 beállításait, újra kell töltenünk annak konfigurációs állományait.

```
root@debian:~# /etc/init.d/apache2 reload
[ ok ] Reloading web server config: apache2.
```

Ezután egy böngészőbe beírva a `www.gyakorlat.com` címet a Rails alkalmazásunk fog megjelenni! Az újabb böngészők kikerülik a `hosts` fájlt a

⁵Alternatív megoldás: `ln -s /etc/apache2/sites-available/gyakorlat.conf /etc/apache2/sites-enabled`

DNS-over-HTTP technológiával, így azt a böngészőben letiltva használhatjuk csak a saját doménnevünket lokálisan. A Rails alkalmazás módosítása után az Apache-beli frissítése a `tmp/restart.txt` fájl időpecsétjének módosításával lehetséges (mentés, létrehozás, `touch` stb.)

Ha a Rails alkalmazás létrehozásakor nem adtunk meg, hogy melyik adatbáziskezelőt kívánjuk használni, 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'
#
default: &default
  adapter: sqlite3
  pool: 5
  timeout: 5000

development:
  <<: *default
  database: db/development.sqlite3

# 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:
  <<: *default
  database: db/test.sqlite3

production:
  <<: *default
  database: db/production.sqlite3
```

Generáljuk újra a projektünket MariaDB/MySQL adatbázis-kezelővel.

```
kovacsg@debian:~> rails new gyakorlat -J -d mysql
```

Az adatbázis megnyitásának két módja azonosos. MySQL esetén, ha nem létezne, előbb létre kell hoznunk a fejlesztői és teszt adatbázisokat.

```
kovacsg@debian:~/gyakorlat> rails db:create
Created database 'gyakorlat_development'
Created database 'gyakorlat_test'
kovacsg@debian:~/gyakorlat> rails db
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 39
Server version: 10.11.3-MariaDB-1 Debian 12

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
MariaDB [gyakorlat_development]> show tables;  
Empty set (0.000 sec)
```

MySQL/MariaDB esetén (`rails new gyakorlat -d mysql`) a `mysql2` adaptert használjuk. A különbség a `Gemfile`-ban és a `database.yml`-ben jelentkezik. Fontos opció az alapértelmezett karakterkódolás megadása, ami, ha lehet hagyjunk változatlanul `utf8mb4` 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/MariaDB esetén szükséges a Rails felhasználó ozzáférési jogosultságainak beállítása, amelyet a MySQL/MariaDB `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.

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. A Rails kontrollerek létrehozásakor automatikusan írja ezt a fájl, de megadhatunk egy általános mintát is a HTTP kérések URI-ainak útvonal részére, 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.

```
Gyakorlat::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 kontrollert a `rails` parancs `generate`, röviden `g` opciójával. A második argumentum (`controller`) azt mondja meg, hogy egy új kontrollert hozunk létre, a harmadik a controller nevét. A negyedik és minden további paraméter a controllerben definiál akciókat. 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.

```
kovacs@debian:~/gyakorlat> rails g
Usage: rails generate GENERATOR [args] [options]

General options:
  -h, [--help]           # Print generator's options and usage
  -p, [--pretend]        # Run but do not make any changes
  -f, [--force]          # Overwrite files that already exist
  -s, [--skip]           # Skip files that already exist
  -q, [--quiet]          # Suppress status output

Please choose a generator below.

Rails:
  application_record
  benchmark
  channel
  controller
  generator
  helper
  integration_test
  jbuilder
  job
  mailbox
  mailer
  migration
  model
  resource
  scaffold
  scaffold_controller
  system_test
  task

ActiveRecord:
  active_record:application_record
  active_record:multi_db

TestUnit:
  test_unit:channel
  test_unit:generator
  test_unit:install
  test_unit:mailbox
  test_unit:plugin
```

```
kovacs@debian:~/gyakorlat> rails g controller
Usage:
  rails generate controller NAME [action action] [options]

Options:
  [--skip-namespace], [--no-skip-namespace] # Skip
  namespace (affects only isolated engines)
  [--skip-collision-check], [--no-skip-collision-check] # Skip
  collision check
  [--skip-routes], [--no-skip-routes] # Don't add
  routes to config/routes.rb.
  [--helper], [--no-helper] # Indicates
  when to generate helper
  -e, [--template-engine=NAME] # Default: true
  # Template
```

```

engine to be invoked

-t, [--test-framework=NAME] # Default: erb
                             # Test
                             # Default:
                             test_unit
                             framework to be invoked

Runtime options:
-f, [--force] # Overwrite files that already exist
-p, [--pretend], [--no-pretend] # Run but do not make any changes
-q, [--quiet], [--no-quiet] # Suppress status output
-s, [--skip], [--no-skip] # Skip files that already exist

Description:
Generates a new controller and its views. Pass the controller name,
either
CamelCased or under_scored, and a list of views as arguments.

To create a controller within a module, specify the controller name as a
path like 'parent_module/controller_name'.

This generates a controller class in app/controllers and invokes helper,
template engine, assets, and test framework generators.

Example:
'bin/rails generate controller CreditCards open debit credit close '

CreditCards controller with URLs like /credit_cards/debit.
Controller: app/controllers/credit_cards_controller.rb
Test:      test/controllers/credit_cards_controller_test.rb
Views:     app/views/credit_cards/debit.html.erb [...]
Helper:    app/helpers/credit_cards_helper.rb
kovacs@debian:~/gyakorlat> rails g 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/controllers/say_controller_test.rb
invoke helper
create app/helpers/say_helper.rb
invoke test_unit

```

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. Ha a `rails s` parancshoz a `-b` kapcsolóval IP címet rendeltünk, akkor a `localhost` helyére

az az IP cím írandó.

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

Ez dinamikussá tehetjük az aktuális idő kiírásával.

```
<%= 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.

```
<%= @time %>
```

A nézet többnyelvűsítését a `/config/locales/` könyvtárban lévő YAML fájlokkal érhetjük el. Definiáljuk a `hello` fordítását és egy időformátumot.

```
en:
  hello: "Hello_world"

  time:
    formats:
      default: "%Y.%m.%d_%H:%M"
      datetime: "%Y.%m.%d._%H:%M:%S"
```

A nézetünkben (`app/views/say/hello.html.erb`) pedig használjuk a fordítást végző `t` (vagy hosszabban `translate`) azonosítójú, és a lokalizációt végző `l` (vagy hosszabban `localize`) azonosítójú függvényt.

```
<h1><%= t 'hello' %></h1>
<p><%= l @time, format: :datetime %></p>
```

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 egységteszteket tartalmazó osztályt és egy tesztadatokat tartalmazó YAML fájlt.

```
kovacs@debian:~/gyakorlat$ rails g model user name:string email:string
neptun:string password:string
invoke active_record
create db/migrate/20231003113429_create_users.rb
create app/models/user.rb
```

```

invoke    test_unit
create    test/models/user_test.rb
create    test/fixtures/users.yml

```

Az adatbázis-migrációs szkriptben az adatmodell változtatásait adjuk meg. Az előző modellgeneráló szkript egy olyan táblát hozna létre, amelyben egy azonosító és két időpecsét attribútum mellett egy `name`, egy `email`, egy `neptun` és egy `password` azonosítójú, string típusú attribútum szerepelne.

```

class CreateUsers < ActiveRecord::Migration[7.0]
  def change
    create_table :users do |t|
      t.string :name
      t.string :email
      t.string :neptun
      t.string :password

      t.timestamps
    end
  end
end

```

Ezután elvégezhetjük a tábla struktúrájának módosítását, amiben 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@debian:~/gyakorlat/db/migrate> rails db:migrate
== 20231003113429 CreateUsers: migrating
-----
-- create_table(:users)
--> 0.0058s
== 20231003113429 CreateUsers: migrated (0.0061s)
-----

```

Az adatbáziskezelővel a `gyakorlat_development` adatbázist kiválasztva ezután ellenőrizhetjük, hogy a táblánk valóban létrejött. Alább ennek ellenőrzése látható a két adatbáziskezelő konzolján.

```

kovacs@debian:~/gyakorlat/db/migrate> rails db

MariaDB [gyakorlat_development]> show tables;
+-----+
| Tables_in_gyakorlat_development |
+-----+
| ar_internal_metadata             |
| schema_migrations                |
| users                             |
+-----+
3 rows in set (0.001 sec)

MariaDB [gyakorlat_development]> desc users;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id         | bigint(20)    | NO   | PRI | NULL    | auto_increment |
| name      | varchar(255)  | YES  |     | NULL    |                |
| email     | varchar(255)  | YES  |     | NULL    |                |
| neptun    | varchar(255)  | YES  |     | NULL    |                |

```

```

+----+-----+-----+-----+-----+-----+
| password | varchar(255) | YES | | NULL | |
| created_at | datetime(6) | NO | | NULL | |
| updated_at | datetime(6) | NO | | NULL | |
+----+-----+-----+-----+-----+
7 rows in set (0.001 sec)

MariaDB [gyakorlat_development]> select * from users;
Empty set (0.000 sec)

```

Nézzük meg, miként tudunk az adatbázishoz Rubyból hozzáférni. Nyisunk egy konzolt, és hozzunk létre egy felhasználót (3. sor), állítsuk be az attribútumait (4-7. sor), majd mentjük el az adatbázisba (8. sor). Nézzük meg ezután, hogy létrejött-e a rekord az adatbázisban, vagyis a példány id attribútuma még mindig `nil`-e (9. sor)?

```

kovacs@debian:~/gyakorlat/db/migrate$ rails c
Loading development environment (Rails 7.0.8)
irb(main):001> User.new
=>
#<User:0x00007f0eac89f5a0
 id: nil,
 name: nil,
 email: nil,
 neptun: nil,
 password: nil,
 created_at: nil,
 updated_at: nil>
irb(main):002> u = User.new
=>
#<User:0x00007f0ead301bb0
 ...
irb(main):003> u.name = "Valaki"
=> "Valaki"
irb(main):004> u.email = "valaki@mail.bme.hu"
=> "valaki@mail.bme.hu"
irb(main):005> u.neptun = "aaaaaa"
=> "aaaaaa"
irb(main):006> u.password = "titok"
=> "titok"
irb(main):007> u
=>
#<User:0x00007f0ead301bb0
 id: nil,
 name: "Valaki",
 email: "valaki@mail.bme.hu",
 neptun: "aaaaaa",
 password: "[FILTERED]",
 created_at: nil,
 updated_at: nil>
irb(main):008> u.save
TRANSACTION (0.1ms) BEGIN
 User Create (2.2ms) INSERT INTO 'users' ('name', 'email', 'neptun', 'password', 'created_at', 'updated_at') VALUES ('Valaki', 'valaki@mail.bme.hu', 'aaaaaa', 'titok', '2023-10-03_11:41:10.695120', '2023-10-03_11:41:10.695120')
TRANSACTION (0.5ms) COMMIT
=> true

```

Ellenőrizzük az adatbáziskonzolon, hogy valóban megjelent-e ez adat az

adatbázisban. Láthatjuk, hogy az időpecsétek automatikusan állítódnak a mentés művelet után.

```
kovacs@debian:~/gyakorlat/db/migrate> rails db
MariaDB [gyakorlat_development]> select * from users;
+-----+-----+-----+-----+-----+-----+
| id | name | email | neptun | password | created_at |
|-----+-----+-----+-----+-----+-----+
| 1 | Valaki | valaki@mail.bme.hu | aaaaaa | titok | 2023-10-03 |
| 11:41:10.695120 | 2023-10-03 11:41:10.695120 |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)
```

Az adatok ezután előkereshetők az adatbázisból. Létrejött a rekord, az azonosítója 1, amit a modell osztály `find` osztálymetódusának átadva az adatbázisból előkereshetjük az objektumot (1. sor). A keresést a `where` metódussal is elvégezhetjük, erről később lesz szó. Az adatok felülírása ismételt `save` vagy az `update` művelettel történik az attribútumok módosítása után.

```
kovacs@debian:~/gyakorlat/db/migrate> rails c
Loading development environment (Rails 7.0.8)
irb(main):001> u = User.find 1
  User Load (0.3ms) SELECT `users`.* FROM `users` WHERE `users`.`id` = 1
  LIMIT 1
=>
#<User:0x00007faeb5351c70
...
irb(main):002> u
=>
#<User:0x00007faeb5351c70
  id: 1,
  name: "Valaki",
  email: "valaki@mail.bme.hu",
  neptun: "aaaaaa",
  password: "[FILTERED]",
  created_at: Tue, 03 Oct 2023 11:41:10.695120000 UTC +00:00,
  updated_at: Tue, 03 Oct 2023 11:41:10.695120000 UTC +00:00>
```

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.