

# A Rails keretrendszer Gyakorlat

Kovács Gábor

2022. szeptember 27.

## 1. Ruby telepítése, Gemek

A Rails keretrendszert egy virtuális környezetben <sup>1</sup> futó Debian Linuxra <sup>2</sup> 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 <sup>3</sup>, 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 webservert is, amelyet az `apt-get install apache2` parancs konzolon történő kiadásával telepíthetünk.

---

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

<sup>2</sup>Mivel 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.

<sup>3</sup>A 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 2.7-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
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libruby2.7 rake ruby-rubygems ruby2.7 ruby2.7-dev
The following NEW packages will be installed:
  libruby2.7 rake ri ruby ruby-dev ruby-rubygems ruby2.7 ruby2.7-dev
0 upgraded, 8 newly installed, 0 to remove and 7 not upgraded.
Need to get 0 B/6,183 kB of archives.
After this operation, 22.4 MB of additional disk space will be used.
Do you want to continue? [Y/n]
Selecting previously unselected package ruby2.7.
(Reading database ... 262322 files and directories currently installed.)
Preparing to unpack .../0-ruby2.7_2.7.4-1+deb11u1_amd64.deb ...
Unpacking ruby2.7 (2.7.4-1+deb11u1) ...
Selecting previously unselected package ruby-rubygems.
Preparing to unpack .../1-ruby-rubygems_3.2.5-2_all.deb ...
Unpacking ruby-rubygems (3.2.5-2) ...
Selecting previously unselected package ruby.
Preparing to unpack .../2-ruby_1%3a2.7+2_amd64.deb ...
Unpacking ruby (1:2.7+2) ...
Selecting previously unselected package rake.
Preparing to unpack .../3-rake_13.0.3-1_all.deb ...
Unpacking rake (13.0.3-1) ...
Selecting previously unselected package libruby2.7:amd64.
Preparing to unpack .../4-libruby2.7_2.7.4-1+deb11u1_amd64.deb ...
Unpacking libruby2.7:amd64 (2.7.4-1+deb11u1) ...
Selecting previously unselected package ri.
Preparing to unpack .../5-ri_1%3a2.7+2_all.deb ...
Unpacking ri (1:2.7+2) ...
Selecting previously unselected package ruby2.7-dev:amd64.
Preparing to unpack .../6-ruby2.7-dev_2.7.4-1+deb11u1_amd64.deb ...
Unpacking ruby2.7-dev:amd64 (2.7.4-1+deb11u1) ...
Selecting previously unselected package ruby-dev:amd64.
Preparing to unpack .../7-ruby-dev_1%3a2.7+2_amd64.deb ...
Unpacking ruby-dev:amd64 (1:2.7+2) ...
Setting up ruby2.7 (2.7.4-1+deb11u1) ...
Setting up ri (1:2.7+2) ...
Setting up ruby (1:2.7+2) ...
Setting up rake (13.0.3-1) ...
Setting up ruby-rubygems (3.2.5-2) ...
Setting up libruby2.7:amd64 (2.7.4-1+deb11u1) ...
Setting up ruby2.7-dev:amd64 (2.7.4-1+deb11u1) ...
Setting up ruby-dev:amd64 (1:2.7+2) ...
Processing triggers for man-db (2.9.4-2) ...
Processing triggers for libc-bin (2.33-7) ...
```

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 2.7.4 p191 (2021-07-07 revision a21a3b7d23) [x86_64-linux-gnu]
root@debian:~# gem list

*** LOCAL GEMS ***

benchmark (default: 0.1.0)
bigdecimal (default: 2.0.0)
bundler (default: 2.1.4)
cgi (default: 0.1.0)
csv (default: 3.1.2)
date (default: 3.0.0)
dbm (default: 1.1.0)
delegate (default: 0.1.0)
did_you_mean (default: 1.4.0)
etc (default: 1.1.0)
fcntl (default: 1.0.0)
fiddle (default: 1.0.0)
fileutils (default: 1.4.1)
forwardable (default: 1.3.1)
gdbm (default: 2.1.0)
getoptlong (default: 0.1.0)
io-console (default: 0.5.6)
ipaddr (default: 1.2.2)
irb (default: 1.2.6)
json (default: 2.3.0)
logger (default: 1.4.2)
matrix (default: 0.2.0)
minitest (5.13.0)
mutex_m (default: 0.1.0)
net-pop (default: 0.1.0)
net-smtp (default: 0.1.0)
net-telnet (0.1.1)
observer (default: 0.1.0)
open3 (default: 0.1.0)
openssl (default: 2.1.2)
ostruct (default: 0.2.0)
power_assert (1.1.7)
prime (default: 0.1.1)
pstore (default: 0.1.0)
psych (default: 3.1.0)
racc (default: 1.4.16)
rake (13.0.3)
rdoc (default: 6.2.1.1)
readline (default: 0.0.2)
readline-ext (default: 0.1.0)
reline (default: 0.1.5)
rexml (default: 3.2.3.1)
rss (default: 0.2.8)
rubygems-update (3.2.5)
sdbm (default: 1.0.0)
singleton (default: 0.1.0)
stringio (default: 0.1.0)
strscan (default: 1.0.3)
test-unit (3.3.9)
timeout (default: 0.1.0)
tracer (default: 0.1.0)
uri (default: 0.10.0)
```

```
webrick (default: 1.6.1)
xmlrpc (0.3.0)
yaml (default: 0.1.0)
zlib (default: 1.1.0)
```

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 tzinfo -2.0.5.gem
Fetching zeitwerk -2.6.0.gem
Fetching i18n -1.12.0.gem
Fetching activesupport -7.0.4.gem
Fetching thor -1.2.1.gem
Fetching nokogiri -1.13.8-x86_64-linux.gem
Fetching method_source -1.0.0.gem
Fetching concurrent-ruby -1.1.10.gem
Fetching crass -1.0.6.gem
Fetching loofah -2.19.0.gem
Fetching rails-html-sanitizer -1.4.3.gem
Fetching rails-dom-testing -2.0.3.gem
Fetching rack -2.2.4.gem
Fetching rack-test -2.0.2.gem
Fetching erubi -1.11.0.gem
Fetching builder -3.2.4.gem
Fetching actionview -7.0.4.gem
Fetching actionpack -7.0.4.gem
Fetching railties -7.0.4.gem
Fetching mini_mime -1.1.2.gem
Fetching marcel -1.0.2.gem
Fetching activemodel -7.0.4.gem
Fetching activerecord -7.0.4.gem
Fetching globalid -1.0.0.gem
Fetching activejob -7.0.4.gem
Fetching activestorage -7.0.4.gem
Fetching actiontext -7.0.4.gem
Fetching net-protocol -0.1.3.gem
Fetching digest -3.1.0.gem
Fetching net-imap -0.2.3.gem
Fetching rails -7.0.4.gem
Fetching actionmailer -7.0.4.gem
Fetching actionmailbox -7.0.4.gem
Fetching websocket-extensions -0.1.5.gem
Fetching websocket-driver -0.7.5.gem
Fetching nio4r -2.5.8.gem
Fetching actioncable -7.0.4.gem
Fetching mail -2.7.1.gem
Successfully installed zeitwerk -2.6.0
Successfully installed thor -1.2.1
Successfully installed method_source -1.0.0
Successfully installed concurrent-ruby -1.1.10
Successfully installed tzinfo -2.0.5
Successfully installed i18n -1.12.0
Successfully installed activesupport -7.0.4
```

```

Successfully installed nokogiri-1.13.8-x86_64-linux
Successfully installed crass-1.0.6
Successfully installed loofah-2.19.0
Successfully installed rails-html-sanitizer-1.4.3
Successfully installed rails-dom-testing-2.0.3
Successfully installed rack-2.2.4
Successfully installed rack-test-2.0.2
Successfully installed erubi-1.11.0
Successfully installed builder-3.2.4
Successfully installed actionview-7.0.4
Successfully installed actionpack-7.0.4
Successfully installed railties-7.0.4
Successfully installed mini_mime-1.1.2
Successfully installed marcel-1.0.2
Successfully installed activemodel-7.0.4
Successfully installed activerecord-7.0.4
Successfully installed globalid-1.0.0
Successfully installed activejob-7.0.4
Successfully installed activestorage-7.0.4
Successfully installed actiontext-7.0.4
Successfully installed net-protocol-0.1.3
Building native extensions. This could take a while...
Successfully installed digest-3.1.0
Successfully installed net-imap-0.2.3
Successfully installed mail-2.7.1
Successfully installed actionmailer-7.0.4
Successfully installed actionmailbox-7.0.4
Successfully installed websocket-extensions-0.1.5
Building native extensions. This could take a while...
Successfully installed websocket-driver-0.7.5
Building native extensions. This could take a while...
Successfully installed nio4r-2.5.8
Successfully installed actioncable-7.0.4
Successfully installed rails-7.0.4
Parsing documentation for zeitwerk-2.6.0
Installing ri documentation for zeitwerk-2.6.0
Parsing documentation for thor-1.2.1
Installing ri documentation for thor-1.2.1
Parsing documentation for method_source-1.0.0
Installing ri documentation for method_source-1.0.0
Parsing documentation for concurrent-ruby-1.1.10
Installing ri documentation for concurrent-ruby-1.1.10
Parsing documentation for tzinfo-2.0.5
Installing ri documentation for tzinfo-2.0.5
Parsing documentation for i18n-1.12.0
Installing ri documentation for i18n-1.12.0
Parsing documentation for activesupport-7.0.4
Installing ri documentation for activesupport-7.0.4
Parsing documentation for nokogiri-1.13.8-x86_64-linux
Installing ri documentation for nokogiri-1.13.8-x86_64-linux
Parsing documentation for crass-1.0.6
Installing ri documentation for crass-1.0.6
Parsing documentation for loofah-2.19.0
Installing ri documentation for loofah-2.19.0
Parsing documentation for rails-html-sanitizer-1.4.3
Installing ri documentation for rails-html-sanitizer-1.4.3
Parsing documentation for rails-dom-testing-2.0.3
Installing ri documentation for rails-dom-testing-2.0.3
Parsing documentation for rack-2.2.4
Installing ri documentation for rack-2.2.4
Parsing documentation for rack-test-2.0.2
Installing ri documentation for rack-test-2.0.2

```

```

Parsing documentation for erubi-1.11.0
Installing ri documentation for erubi-1.11.0
Parsing documentation for builder-3.2.4
Installing ri documentation for builder-3.2.4
Parsing documentation for actionview-7.0.4
Installing ri documentation for actionview-7.0.4
Parsing documentation for actionpack-7.0.4
Installing ri documentation for actionpack-7.0.4
Parsing documentation for railties-7.0.4
Installing ri documentation for railties-7.0.4
Parsing documentation for mini_mime-1.1.2
Installing ri documentation for mini_mime-1.1.2
Parsing documentation for marcel-1.0.2
Installing ri documentation for marcel-1.0.2
Parsing documentation for activemodel-7.0.4
Installing ri documentation for activemodel-7.0.4
Parsing documentation for activerecord-7.0.4
Installing ri documentation for activerecord-7.0.4
Parsing documentation for globalid-1.0.0
Installing ri documentation for globalid-1.0.0
Parsing documentation for activejob-7.0.4
Installing ri documentation for activejob-7.0.4
Parsing documentation for activestorage-7.0.4
Installing ri documentation for activestorage-7.0.4
Parsing documentation for actiontext-7.0.4
Installing ri documentation for actiontext-7.0.4
Parsing documentation for net-protocol-0.1.3
Installing ri documentation for net-protocol-0.1.3
Parsing documentation for digest-3.1.0
Installing ri documentation for digest-3.1.0
Parsing documentation for net-imap-0.2.3
Installing ri documentation for net-imap-0.2.3
Parsing documentation for mail-2.7.1
Installing ri documentation for mail-2.7.1
Parsing documentation for actionmailer-7.0.4
Installing ri documentation for actionmailer-7.0.4
Parsing documentation for actionmailbox-7.0.4
Installing ri documentation for actionmailbox-7.0.4
Parsing documentation for websocket-extensions-0.1.5
Installing ri documentation for websocket-extensions-0.1.5
Parsing documentation for websocket-driver-0.7.5
Installing ri documentation for websocket-driver-0.7.5
Parsing documentation for nio4r-2.5.8
Installing ri documentation for nio4r-2.5.8
Parsing documentation for actioncable-7.0.4
Installing ri documentation for actioncable-7.0.4
Parsing documentation for rails-7.0.4
Installing ri documentation for rails-7.0.4
Done installing documentation for zeitwerk, thor, method_source, concurrent-
  ruby, tzinfo, 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, net-
  protocol, digest, net-imap, mail, actionmailer, actionmailbox, websocket
  -extensions, websocket-driver, nio4r, actioncable, rails after 36
  seconds
38 gems installed

```

A Rails rendszerünk konfigurációja a következő témánk. Kétféle web-  
szerveret használunk, a fejlesztéshez a Puma-t, míg az éles rendszerhez az

Apache2-t. Adatbáziskezelőből szintén kétfélet 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 webserverral való integrációt lehetővé tevő Passenger plugin.

```
root@debian:~# gem install passenger
Fetching passenger-6.0.15.gem
Building native extensions. This could take a while...
Successfully installed passenger-6.0.15
Parsing documentation for passenger-6.0.15
Installing ri documentation for passenger-6.0.15
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 libsqlite3-dev
Reading package lists... Done
Building dependency tree... Done
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 7 not upgraded.
Need to get 963 kB of archives.
After this operation, 2,620 kB of additional disk space will be used.
Get:1 http://ftp.hu.debian.org/debian stable/main amd64 libsqlite3-dev amd64
3.39.3-1 [1,027 kB]
Fetched 963 kB in 0s (2,813 kB/s)
Selecting previously unselected package libsqlite3-dev:amd64.
(Reading database ... 225987 files and directories currently installed.)
Preparing to unpack .../libsqlite3-dev_3.39.3-1_amd64.deb ...
Unpacking libsqlite3-dev:amd64 (3.39.3-1) ...
Setting up libsqlite3-dev:amd64 (3.39.3-1) ...
```

```
root@debian:~# apt-get install libmysqlclient-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libmysqlclient21 libzstd-dev
The following NEW packages will be installed:
  libmysqlclient-dev libmysqlclient21 libzstd-dev
0 upgraded, 3 newly installed, 1 to remove and 950 not upgraded.
Need to get 2,927 kB of archives.
After this operation, 16.5 MB of additional disk space will be used.
Do you want to continue? [Y/n]
Get:1 http://ftp.hu.debian.org/debian sid/main amd64 libmysqlclient21 amd64
8.0.30-1 [1,190 kB]
Get:2 http://ftp.hu.debian.org/debian sid/main amd64 libzstd-dev amd64
1.5.2+dfsg-1 [333 kB]
```

```

Get:3 http://ftp.hu.debian.org/debian sid/main amd64 libmysqlclient-dev
amd64 8.0.30-1 [1,463 kB]
Get:1 http://ftp.hu.debian.org/debian sid/main amd64 libmysqlclient21 amd64
8.0.23-3+b1 [1,149 kB]
Fetched 2,927 kB in 1s (3,442 kB/s)
(Reading database ... 225995 files and directories currently installed.)
Selecting previously unselected package libmysqlclient21:amd64.
Preparing to unpack .../libmysqlclient21_8.0.30-1_amd64.deb ...
Unpacking libmysqlclient21:amd64 (8.0.30-1) ...
Selecting previously unselected package libzstd-dev:amd64.
Preparing to unpack .../libzstd-dev_1.5.2+dfsg-1_amd64.deb ...
Unpacking libzstd-dev:amd64 (1.5.2+dfsg-1) ...
Selecting previously unselected package libmysqlclient-dev.
Preparing to unpack .../libmysqlclient-dev_8.0.30-1_amd64.deb ...
Unpacking libmysqlclient-dev (8.0.30-1) ...
Setting up libzstd-dev:amd64 (1.5.2+dfsg-1) ...
Setting up libmysqlclient21:amd64 (8.0.30-1) ...
Setting up libmysqlclient-dev (8.0.30-1) ...

```

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:~/pluto# gem install sqlite3
Fetching sqlite3-1.5.0-x86_64-linux.gem
Successfully installed sqlite3-1.5.0-x86_64-linux
Parsing documentation for sqlite3-1.5.0-x86_64-linux
Installing ri documentation for sqlite3-1.5.0-x86_64-linux
Done installing documentation for sqlite3 after 1 seconds
1 gem installed

```

```

root@debian:~/pluto# gem install mysql2
Fetching mysql2-0.5.4.gem
Building native extensions. This could take a while...
Successfully installed mysql2-0.5.4
Parsing documentation for mysql2-0.5.4
Installing ri documentation for mysql2-0.5.4
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.

```

root@debian:~/# gem install bundler
Fetching bundler-2.3.22.gem
Successfully installed bundler-2.3.22
Parsing documentation for bundler-2.3.22
Installing ri documentation for bundler-2.3.22
Done installing documentation for bundler after 0 seconds
1 gem installed

```



## 2. Rails alkalmazás létrehozása

A Rails használatához a rails szkriptet használjuk immáron nem rendszergazdaként, hanem egyszerű felhasználóként. A parancs Debian/Ubuntu Linux és 2.7-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.

```
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 webservert. 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 pluto -d mysql -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/pluto/.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 szerializált 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 fájlokat, mint például `session` azonosítókat, sütiket.

A `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 környezetben vagy a már említett `lib` és `vendor` könyvtárakban érhetők 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` min-

den 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.<sup>4</sup>

```
kovacs@debian:~/pluto> 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/2.7.0
  /var/lib/gems/2.7.0/build_info
  /var/lib/gems/2.7.0/cache
  /var/lib/gems/2.7.0/doc
  /var/lib/gems/2.7.0/extensions
  /var/lib/gems/2.7.0/gems
  /var/lib/gems/2.7.0/plugins
  /var/lib/gems/2.7.0/specifications
Using concurrent-ruby 1.1.10
Using i18n 1.12.0
Using minitest 5.16.3
Using tzinfo 2.0.5
Using activesupport 7.0.4
Using builder 3.2.4
Using erubi 1.11.0
Using racc 1.6.0
Using nokogiri 1.13.8 (x86_64-linux)
Using rails-dom-testing 2.0.3
Using crass 1.0.6
Using loofah 2.19.0
Using rails-html-sanitizer 1.4.3
Using actionview 7.0.4
Using rack 2.2.4
Using rack-test 2.0.2
Using actionpack 7.0.4
Using nio4r 2.5.8
Using websocket-extensions 0.1.5
Using websocket-driver 0.7.5
Using actioncable 7.0.4
Using globalid 1.0.0
Using activejob 7.0.4
Using activemodel 7.0.4
Using activerecord 7.0.4
Using marcel 1.0.2
```

<sup>4</sup>A 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.

```

Using mini_mime 1.1.2
Using activestorage 7.0.4
Using mail 2.7.1
Using digest 3.1.0
Using timeout 0.3.0
Using net-protocol 0.1.3
Using strscan 3.0.4
Using net-imap 0.2.3
Using net-pop 0.1.1
Using net-smtp 0.3.1
Using actionmailbox 7.0.4
Using actionmailer 7.0.4
Using actiontext 7.0.4
Using public_suffix 5.0.0
Using addressable 2.8.1
Using bindex 0.8.1
Using msgpack 1.5.6
Using bootsnap 1.13.0
Using bundler 2.3.22
Using matrix 0.4.2
Using regexp_parser 2.6.0
Using xpath 3.2.0
Using capybara 3.37.1
Using childprocess 4.1.0
Using io-console 0.5.11
Using reline 0.3.1
Using irb 1.4.1
Using debug 1.6.2
Using method_source 1.0.0
Using thor 1.2.1
Using zeitwerk 2.6.0
Using railties 7.0.4
Using importmap-rails 1.1.5
Using jbuilder 2.11.5
Using mysql2 0.5.4
Using puma 5.6.5
Using rails 7.0.4
Using rexml 3.2.5
Using rubyzip 2.3.2
Using websocket 1.2.9
Using selenium-webdriver 4.4.0
Using sprockets 4.1.1
Using sprockets-rails 3.4.2
Using stimulus-rails 1.1.0
Using turbo-rails 1.3.0
Using web-console 4.2.0
Using webdrivers 5.1.0
Bundle complete! 15 Gemfile dependencies, 74 gems now installed.
Use 'bundle info [gemname]' to see where a bundled gem is installed.

```

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.

```

kovacsg@debian:~/pluto> gem list

*** LOCAL GEMS ***

actioncable (7.0.4)
actionmailbox (7.0.4)
actionmailer (7.0.4)

```

```
actionpack (7.0.4)
actiontext (7.0.4)
actionview (7.0.4)
activejob (7.0.4)
activemodel (7.0.4)
activerecord (7.0.4)
activestorage (7.0.4)
activesupport (7.0.4)
addressable (2.8.1)
benchmark (default: 0.1.0)
bigdecimal (default: 2.0.0)
bindex (0.8.1)
bootsnap (1.13.0)
builder (3.2.4)
bundler (2.3.22, default: 2.1.4)
capybara (3.37.1)
cgi (default: 0.1.0)
childprocess (4.1.0)
concurrent-ruby (1.1.10)
crass (1.0.6)
csv (default: 3.1.2)
date (default: 3.0.0)
dbm (default: 1.1.0)
debug (1.6.2)
delegate (default: 0.1.0)
did_you_mean (default: 1.4.0)
digest (3.1.0)
erubi (1.11.0)
etc (default: 1.1.0)
fcntl (default: 1.0.0)
fiddle (default: 1.0.0)
fileutils (default: 1.4.1)
forwardable (default: 1.3.1)
gdbm (default: 2.1.0)
getoptlong (default: 0.1.0)
globalid (1.0.0)
i18n (1.12.0)
importmap-rails (1.1.5)
io-console (0.5.11, default: 0.5.6)
ipaddr (default: 1.2.2)
irb (1.4.1, default: 1.2.6)
jbuilder (2.11.5)
json (default: 2.3.0)
logger (default: 1.4.2)
loofah (2.19.0)
mail (2.7.1)
marcel (1.0.2)
matrix (0.4.2, default: 0.2.0)
method_source (1.0.0)
mini_mime (1.1.2)
minitest (5.16.3, 5.13.0)
msgpack (1.5.6)
mutex_m (default: 0.1.0)
mysql2 (0.5.4)
net-imap (0.2.3)
net-pop (0.1.1, default: 0.1.0)
net-protocol (0.1.3)
net-smtp (0.3.1, default: 0.1.0)
net-telnet (0.1.1)
nio4r (2.5.8)
nokogiri (1.13.8 x86_64-linux)
observer (default: 0.1.0)
```



```

open3 (default: 0.1.0)
openssl (default: 2.1.2)
ostruct (default: 0.2.0)
passenger (6.0.15)
power_assert (1.1.7)
prime (default: 0.1.1)
pstore (default: 0.1.0)
psych (default: 3.1.0)
public_suffix (5.0.0)
puma (5.6.5)
racc (1.6.0, default: 1.4.16)
rack (2.2.4)
rack-test (2.0.2)
rails (7.0.4)
rails-dom-testing (2.0.3)
rails-html-sanitizer (1.4.3)
railties (7.0.4)
rake (13.0.6, 13.0.3)
rdoc (default: 6.2.1.1)
readline (default: 0.0.2)
readline-ext (default: 0.1.0)
regexp_parser (2.6.0)
reline (0.3.1, default: 0.1.5)
rexml (3.2.5, default: 3.2.3.1)
rss (default: 0.2.8)
rubygems-update (3.2.5)
rubyzip (2.3.2)
sdbm (default: 1.0.0)
selenium-webdriver (4.4.0)
singleton (default: 0.1.0)
sprockets (4.1.1)
sprockets-rails (3.4.2)
sqlite3 (1.5.0 x86_64-linux)
stimulus-rails (1.1.0)
stringio (default: 0.1.0)
strscan (3.0.4, default: 1.0.3)
test-unit (3.3.9)
thor (1.2.1)
timeout (0.3.0, default: 0.1.0)
tracer (default: 0.1.0)
turbo-rails (1.3.0)
tzinfo (2.0.5)
uri (default: 0.10.0)
web-console (4.2.0)
webdrivers (5.1.0)
webrick (default: 1.6.1)
websocket (1.2.9)
websocket-driver (0.7.5)
websocket-extensions (0.1.5)
xmlrpc (0.3.0)
xpath (3.2.0)
yaml (default: 0.1.0)
zeitwerk (2.6.0)
zlib (default: 1.1.0)

```

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:~/pluto> 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
importmap:install
initializers
log:clear
middleware
notes
restart
routes
runner
secret
secrets:edit
secrets:setup
secrets:show
stats
stimulus:install
stimulus:install:importmap
stimulus:install:node
test:all
test:db
time:zones[country_or_offset]
tmp:clear
tmp:create
turbo:install
turbo:install:importmap
turbo:install:node
turbo:install:redis
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:~/pluto> rails c
Loading development environment (Rails 7.0.4)
irb(main):001:0> Rails.application
=> #<Pluto::Application>
irb(main):002:0>
```

### 3. Web- és adatbázisszerver kapcsolat beállítása

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

Az alapértelmezett webszervert, 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:~/pluto$ rails s -b 10.211.55.3
=> Booting Puma
=> Rails 7.0.4 application starting in development
=> Run 'bin/rails server --help' for more startup options
Puma starting in single mode...
* Puma version: 5.6.5 (ruby 2.7.4-p191) ("Birdie's Version")
* Min threads: 5
* Max threads: 5
* Environment: development
* PID: 79654
* 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, v6.0.15.

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 7.74.0
  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: 2.4.52
* Checking for Rake (associated with /usr/bin/ruby3.0) ...
  Found: yes
  Location: /usr/bin/ruby3.0 /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-3.0.0/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: 1.7.0
* Checking for Apache Portable Runtime Utility (APU) development headers...
  Found: yes
  Location: /usr/bin/apu-1-config
  Version: 1.6.1
```

```
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/2.7.0/gems/passenger-6.0.15/
  buildout/apache2/mod_passenger.so
<IfModule mod_passenger.c>
  PassengerRoot /var/lib/gems/2.7.0/gems/passenger-6.0.15
  PassengerDefaultRuby /usr/bin/ruby2.7
</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 kiírja a konzolra. A fájlt /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/2.7.0/gems/passenger-6.0.15/
  buildout/apache2/mod_passenger.so
<IfModule mod_passenger.c>
  PassengerRoot /var/lib/gems/2.7.0/gems/passenger-6.0.15
```

```
PassengerDefaultRuby /usr/bin/ruby2.7
</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 oldalt<sup>5</sup>. 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/kovacsg/pluto/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/kovacsg/pluto/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 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 konfigurá-

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

ciós 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
```

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

```
kovacs@debian:~/pluto/config> rails db:create
Created database 'pluto_development'
Created database 'pluto_test'
kovacs@debian:~/pluto/config> rails db
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 46
Server version: 10.6.7-MariaDB-2 Debian buildd-unstable

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

MySQL/MariaDB esetén (`rails new gyakorlat -d mysql`) a `mysql2` adaptort 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.

A MySQL/MariaDB adatbázis nem jön automatikusan létre az SQLite3-mal ellentétben. A létrehozása következő parancsokkal történhet meg. Először eltávolítjuk az összes azonos nevű adatbázist a `db:drop` opcióval, majd a `db:create` opciót használjuk, amely a `development` és a `test` környezetekhez hozza létre egy üres sémát. Az eredmény a Rails adatbázis konzolán ellenőrizzük.

```
kovacs@debian:~/pluto/config> rails db:create
Created database 'pluto_development'
Created database 'pluto_test'
kovacs@debian:~/pluto/config> rails db:drop
Dropped database 'pluto_development'
Dropped database 'pluto_test'
```

## 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 akciót. E parancs

négy Ruby forrásfájlt és egy könyvárat hoz létre az akcióknak megfelelő web-oldalak, view-k számára. A controller nevének megfelelő controller osztályt (`say_controller.rb`), helper osztályt, illetve ezek funkcionális és egységtesztjéhez használható osztályokat.

```
kovacs@debian:~/pluto> 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

Stimulus:
  stimulus

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

```
kovacs@debian:~/pluto> 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
        engine to be invoked                 # Template

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

                                           # Default:
                                           test_unit

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:~/pluto/config> 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
      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 oszlály 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:~/pluto/config> rails g model user name:string email:string
password:string pluto:string
  invoke  active_record
  create  db/migrate/20220927113845_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 `password`, és egy `pluto` 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 :password
      t.string :pluto

      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:~/pluto/config> rails db:migrate
== 20220927113845 CreateUsers: migrating
-----
-- create_table(:users)
--> 0.0066s
== 20220927113845 CreateUsers: migrated (0.0068s)
-----

```

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

```

kovacs@debian:~/pluto/config> rails db
MariaDB [pluto_development]> show tables;
+-----+
| Tables_in_pluto_development |
+-----+
| ar_internal_metadata         |
| schema_migrations           |
| users                        |
+-----+
3 rows in set (0.000 sec)

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

MariaDB [pluto_development]> select * from users;
Empty set (0.001 sec)

```

Nézzük meg, miként tudunk az adatbázishoz Rubyből hozzáférni. Nyissunk egy konzolt, és hozzunk létre egy felhasználót (2. sor), állítsuk be az attribútumait (3-6. sor), vizsgáljuk meg az objektumot (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:~/pluto/config> rails c
Loading development environment (Rails 7.0.4)
irb(main):001:0> User
=> User (call 'User.connection' to establish a connection)
irb(main):002:0> u = User.new
=>
#<User:0x00005644937340c8
...
irb(main):003:0> u.name = 'Senki'
=> "Senki"
irb(main):004:0> u.email = 'senki@mail.bme.hu'
=> "senki@mail.bme.hu"
irb(main):005:0> u.password = 'titok'
=> "titok"
irb(main):006:0> u.pluto = 'aaaaaa'
=> "aaaaaa"
irb(main):007:0> u
=>
#<User:0x00005644937340c8
id: nil,
name: "Senki",
email: "senki@mail.bme.hu",
password: "[FILTERED]",
pluto: "aaaaaa",
created_at: nil,
updated_at: nil>
irb(main):008:0> u.save
TRANSACTION (0.2ms) BEGIN
User Create (0.5ms) INSERT INTO 'users' ('name', 'email', 'password', 'pluto', 'created_at', 'updated_at') VALUES ('Senki', 'senki@mail.bme.hu', 'titok', 'aaaaaa', '2022-09-27_11:44:24.300391', '2022-09-27_11:44:24.300391')
TRANSACTION (0.5ms) COMMIT
=> true
irb(main):009:0> u
=>
#<User:0x00005644937340c8
id: 1,
name: "Senki",
email: "senki@mail.bme.hu",

```

```
password: "[FILTERED]",
pluto: "aaaaaa",
created_at: Tue, 27 Sep 2022 11:44:24.300391000 UTC +00:00,
updated_at: Tue, 27 Sep 2022 11:44:24.300391000 UTC +00:00>
```

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

```
kovacs@debian:~/pluto/config> rails db
MariaDB [pluto_development]> select * from users;
```

id	name	email	password	pluto	created_at
1	Senki	senki@mail.bme.hu	titok	aaaaaa	2022-09-27 11:44:24.300391

Az adatok eztán 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:~/pluto/config> rails c
Loading development environment (Rails 7.0.4)
irb(main):001:0> u = User.find 1
  User Load (0.2ms)  SELECT 'users'.* FROM 'users' WHERE 'users'. 'id' = 1
  LIMIT 1
=>
#<User:0x00007f960083e9b8
...
irb(main):002:0> u
=>
#<User:0x00007f960083e9b8
 id: 1,
 name: "Senki",
 email: "senki@mail.bme.hu",
 password: "[FILTERED]",
 pluto: "aaaaaa",
 created_at: Tue, 27 Sep 2022 11:44:24.300391000 UTC +00:00,
 updated_at: Tue, 27 Sep 2022 11:44:24.300391000 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.