

## ruby - Nove funkcije #18004

### TDD, BDD: cucumber, spork test server

25.07.2009 20:53 - Ernad Husremović

<b>Status:</b> Zatvoreno	<b>Početak:</b> 25.07.2009
<b>Prioritet:</b> Normalan	<b>Završetak:</b>
<b>Odgovorna osoba:</b> Ernad Husremović	<b>% završeno:</b> 0%
<b>Kategorija:</b>	<b>Procjena vremena:</b> 0.00 sat
<b>Ciljna verzija:</b>	
<b>Opis</b> bdd cucumber	
<b>Povezani tiketi:</b> korelira sa ruby - Podrška #18003: require ruby gem specific version <span style="float: right;"><b>Zatvoreno</b> <b>25.07.2009</b></span>	

### Historija

#### #1 - 25.07.2009 20:55 - Ernad Husremović

bringout@nmraka-2:~/devel/git/upstream/carrierwave\$ rake features

```
(in /home/bringout/devel/git/upstream/carrierwave)
/usr/bin/ruby1.8 -I "/home/bringout/.gem/ruby/1.8/gems/cucumber-0.3.90/lib:lib" "/home/bringout/.gem/ruby/1.8/gems/cucumber-0.3.90/bin/cucumber" --profile default features/versions_overridden_filename.feature features/mount_datamapper.feature features/file_storage_reversing_processor.feature features/file_storage_overridden_filename.feature features/versions_overridden_store_dir.feature features/caching.feature features/file_storage.feature features/file_storage_overridden_store_dir.feature features/versions_nested_versions.feature features/versions_basics.feature features/mount_activerecord.feature
```

```
...
42 scenarios (42 passed)
428 steps (428 passed)
0m2.085s
```

jedan feature:

bringout@nmraka-2:~/devel/git/upstream/carrierwave\$ cucumber features/mount\_activerecord.feature

Feature: Mount an Uploader on ActiveRecord class

In order to easily attach files to a form  
As a web developer using CarrierWave  
I want to mount an uploader on an ActiveRecord class

```
Background: # features/mount_activerecord.feature:6
  Given an uploader class that uses the 'file' storage # features/step_definitions/general_steps.rb:3
  And an activerecord class that uses the 'users' table # features/step_definitions/activerecord_steps.rb:3
:3
  And the uploader class is mounted on the 'avatar' column # features/step_definitions/mount_steps.rb:7
  And an instance of the activerecord class # features/step_definitions/activerecord_steps.rb:8
:8
```

```
Scenario: assign a file # features/mount_activerecord.feature:12
  When I assign the file 'fixtures/bork.txt' to the 'avatar' column # features/step_definitions/mount_steps.rb:3
  Then there should be a file called 'bork.txt' somewhere in a subdirectory of 'public/uploads/tmp' # features/step_definitions/file_steps.rb:14
  And the file called 'bork.txt' in a subdirectory of 'public/uploads/tmp' should be identical to the file at 'fixtures/bork.txt' # features/step_definitions/file_steps.rb:29
```

```
Scenario: assign a file and save the record # feature
s/mount_activerecord.feature:17
  When I assign the file 'fixtures/bork.txt' to the 'avatar' column # feature
s/step_definitions/mount_steps.rb:3
  And I save the active record # feature
s/step_definitions/activerecord_steps.rb:12
  Then there should be a file at 'public/uploads/bork.txt' # feature
```

```

s/step_definitions/file_steps.rb:6
  And the file at 'public/uploads/bork.txt' should be identical to the file at 'fixtures/bork.txt' # feature
s/step_definitions/file_steps.rb:21
  And the url for the column 'avatar' should be '/uploads/bork.txt' # feature
s/step_definitions/mount_steps.rb:17

Scenario: assign a file and retrieve it from cache # feature
s/mount_activerecord.feature:24
  When I assign the file 'fixtures/bork.txt' to the 'avatar' column # feature
s/step_definitions/mount_steps.rb:3
  And I retrieve the file later from the cache name for the column 'avatar' # feature
s/step_definitions/mount_steps.rb:11
  And I save the active record # feature
s/step_definitions/activerecord_steps.rb:12
  Then there should be a file at 'public/uploads/bork.txt' # feature
s/step_definitions/file_steps.rb:6
  And the file at 'public/uploads/bork.txt' should be identical to the file at 'fixtures/bork.txt' # feature
s/step_definitions/file_steps.rb:21
  And the url for the column 'avatar' should be '/uploads/bork.txt' # feature
s/step_definitions/mount_steps.rb:17

Scenario: store a file and retrieve it later # features/mount_activerecord.fe
ature:32
  When I assign the file 'fixtures/bork.txt' to the 'avatar' column # features/step_definitions/moun
t_steps.rb:3
  And I retrieve the file later from the cache name for the column 'avatar' # features/step_definitions/moun
t_steps.rb:11
  And I save the active record # features/step_definitions/acti
verecord_steps.rb:12
  Then there should be a file at 'public/uploads/bork.txt' # features/step_definitions/file
_steps.rb:6
  When I reload the active record # features/step_definitions/acti
verecord_steps.rb:16
  Then the url for the column 'avatar' should be '/uploads/bork.txt' # features/step_definitions/moun
t_steps.rb:17

Scenario: store a file and delete the record # features/mount_activerecord.fe
ature:40
  When I assign the file 'fixtures/bork.txt' to the 'avatar' column # features/step_definitions/moun
t_steps.rb:3
  And I retrieve the file later from the cache name for the column 'avatar' # features/step_definitions/moun
t_steps.rb:11
  And I save the active record # features/step_definitions/acti
verecord_steps.rb:12
  Then there should be a file at 'public/uploads/bork.txt' # features/step_definitions/file
_steps.rb:6
CarrierWave: removing file
  When I delete the active record # features/step_definitions/acti
verecord_steps.rb:20
  Then there should not be a file at 'public/uploads/bork.txt' # features/step_definitions/file
_steps.rb:10

5 scenarios (5 passed)
46 steps (46 passed)
0m0.796s

```

#2 - 25.07.2009 20:58 - Ernad Husremović

## od čega se to sastoji

### features fajlovi - opisi onoga što želimo dobiti

```
bringout@nmraka-2:~/devel/git/upstream/carrierwave$ ls features
```

```

caching.feature          file_storage_reversing_processor.feature  step_definitions
  versions_overridden_filename.feature
file_storage.feature     fixtures                                  support
  versions_overridden_store_dir.feature
file_storage_overridden_filename.feature  mount_activerecord.feature              versions_basics.feature
file_storage_overridden_store_dir.feature  mount_datamapper.feature                versions_nested_versions.
feature

```

### definicije pojedinih koraka - steps

```
bringout@nmraka-2:~/devel/git/upstream/carrierwave$ ls features/step_definitions/
```

```
activerecord_steps.rb  caching_steps.rb  datamapper_steps.rb  file_steps.rb  general_steps.rb  mount_steps.rb  
store_steps.rb
```

## klase rade konkretne funkcije

```
bringout@nmraka-2:~/devel/git/upstream/carrierwave$ ls features/support
```

```
activerecord.rb  datamapper.rb  env.rb
```

## testni podaci

```
bringout@nmraka-2:~/devel/git/upstream/carrierwave$ ls features/fixtures  
bork.txt  monkey.txt
```

### #3 - 26.07.2009 10:38 - Ernad Husremović

- *Fajl Cucumber\_Driven\_Classes\_5\_11\_qtob.mov dodano*

uz pomoć ovog 3.5minutnog videa fino sam skontao cucumber, pa sam na osnovu njega

napravio svoj [cucumber lab](#)

u lab-u sam postupno (iz više commit-a) došao do finih rezultata:

1. kako se kreiraju step-ovi (cucumber mi pomaže)
2. kako se postupno implementira ono što sa `car.feature` i `car_steps.rb` očekujem
3. i na kraju [refactoring](#)

Tom prilikom sam još jednom posvjedočio kako je ruby cool language

### #4 - 13.04.2010 14:05 - Ernad Husremović

- *Naslov promijenjeno iz cucumber u TDD, BDD: cucumber, spork test server*

## spork test server

<http://github.com/timcharper/spork>

Spork is Tim Harper's implementation of test server (similar to the `script/spec_server` provided by `rspec-rails`), except rather than using the Rails constant unloading to reload your files, it forks a copy of the server each time you run your tests. The result? Spork runs more solid: it doesn't get corrupted over time, and it properly handles modules and any voo-doo meta programming you may have put in your app.

Spork runs on POSIX systems using fork. It also runs on windows by pre-populating a pool of ready processes (referred to here as the "magazine" strategy).

Supported Testing Frameworks

- Rspec
- Cucumber
- Test::Unit (via `'gem install spork-testunit'`)

### #5 - 26.05.2010 15:11 - Ernad Husremović

- *Status promijenjeno iz Dodijeljeno u Zatvoreno*

## Fajlovi

---

Cucumber_Driven_Classes_5_11_qtob.mov	31,6 MB	26.07.2009	Ernad Husremović
---------------------------------------	---------	------------	------------------