

[My Bread](#)

Software and Other Thoughts

- [RSS](#)

| |
|--|
| <input type="text" value="Search"/> |
| <input type="button" value="Navigate... ▼"/> |

- [Blog](#)
- [Archives](#)

Basic Testing Setup With RSpec and FactoryGirl

Feb 4th, 2014

tl;dr

- Hate TDD blah blah
- A directory structure of a Ruby program
- Setting up RSpec
- Setting up FactoryGirl

Setting up RSpec and FactoryGirl wanted me to choke a few people.

— *John Smith*, renown Rubyist, husband, author of no books, father of none

An Adaquate Response by @tenderlove

**Aaron Patterson**

@tenderlove

[Follow](#)

Here is a Friday Hug. Deal with it. (HAPPY FRIDAY!!! <3<3<3)

7:12 PM - 24 Jan 2014

3 16

O boy, do I hate test driven development. Nothing feels worse than writing clean automated tests only to realize that they've just rendered useless by a small change in an application feature. And yet, we writes tests anyway because they pay off in the long run.

With that in mind, I wanted to write a guide on preparing RSpec and FactoryGirl for writing Ruby programs. RSpec and FactoryGirl are great Gems that help you immensely in writing automated tests, but I've had times when I would curse gods in just prepping RSpec and FactoryGirl.

A Directory Structure of a Ruby Program

There is no *one right way* to structure a Ruby program. However, from my experience, I notice myself increasingly structuring Ruby programs as if I am writing a Ruby Gem. Others looking at the code understand the program better, if they are familiar with how Ruby Gems are typically written. Modularized code helped me debug easier as well.

Let's say we are writing a Ruby program for a linked list with a List class and a Node class. Then we will have a directory structure like below.

```
├── Gemfile
├── Gemfile.lock
├── lib
│   ├── models
│   │   ├── list.rb
│   │   └── node.rb
├── spec
│   ├── factories
│   │   ├── lists.rb
│   │   └── nodes.rb
│   ├── models
│   │   ├── list_spec.rb
│   │   └── node_spec.rb
└── spec_helper.rb
```

- Gemfile: contains a list of Gems used in the program
- lib: a directory that contains code that runs the program

- spec: a directory that contains RSpec and FactoryGirl related code
- spec/factories: a directory that contains factories for FactoryGirl

Setting Up RSpec and FactoryGirl

Tutorial Environment

- Mac OS X Mavericks Version 10.9.1
- ruby 2.1.0p0 (2013-12-25 revision 44422) [x86_64-darwin12.0]
- rspec 2.14.7

1. Gemfile

Prepare Gemfile with RSpec and FactoryGirl.

Gemfile

```
1 source 'https://rubygems.org'
2
3 gem "rspec"
4 gem "factory_girl"
```

Run `$ bundle install`.

2. Prepare RSpec

RSpec is a Ruby Gem that helps you write automated tests.

spec/spec_helper.rb

```
1 require 'rubygems'
2
3 RSpec.configure do |config|
4 end
```

3. Prepare FactoryGirl

FactoryGirl is a Ruby Gem that mocks certain classes for automated tests. We will use FactoryGirl to mock nodes and lists.

Add this to the RSpec config.

spec/spec_helper.rb

```
1 require 'rubygems'
2 require 'factory_girl'
3
4 RSpec.configure do |config|
5   config.include FactoryGirl::Syntax::Methods
6 end
```

4. Building Factories

These are just empty factories.

spec/factories/lists.rb

```
1 FactoryGirl.define do
2   factory :list do
3     end
4 end
```

```
4 end
```

```
spec/factories/nodes.rb
```

```
1 FactoryGirl.define do
2   factory :node do
3   end
4 end
```

5. Describing Models in RSpec

And these are just empty model specs.

```
spec/models/node_spec.rb
```

```
1 require 'spec_helper'
2
3 describe Node do
4
5 end
```

```
spec/models/list_spec.rb
```

```
1 require 'spec_helper'
2
3 describe List do
4
5 end
```

Try running RSpec tests now. Run `$ rspec spec`.

You should get an error message that reads something like:

```
/...../spec/models/list_spec.rb:3:in `<<top (required)>': uninitialized constant List (NameError)
```

6. Add Models

The error above is caused by the fact that we don't actually have any models yet.

```
lib/models/node.rb
```

```
1 class Node
2 end
```

```
lib/models/list.rb
```

```
1 class List
2 end
```

Try running RSpec tests again. Run `$ rspec spec`.

You should get the same error message that reads:

```
/...../spec/models/list_spec.rb:3:in `<<top (required)>': uninitialized constant List (NameError)
```

7. Adding Dependancies in RSpec Config

We are getting this error because RSpec is unable to location the models we just created.

```
spec/spec_helper
```

```

1 require 'rubygems'
2 require 'factory_girl'
3
4 require_relative '../lib/models/list'
5 require_relative '../lib/models/node'
6
7 RSpec.configure do |config|
8   config.include FactoryGirl::Syntax::Methods
9 end

```

Now try running `$ rspec spec`.

Voila! Tests are running! Well, we don't really have any tests, but we have RSpec finally running.

No examples found.

```

Finished in 0.00006 seconds
0 examples, 0 failures

```

8. Running Tests

We will write very basic test that confirms that we can create a Node object and a List object.

spec/models/node_spec.rb

```

1 require 'spec_helper'
2
3 describe Node do
4   describe 'instantiation' do
5     let!(:node) { build(:node) }
6
7     it 'instantiates a list' do
8       expect(node.class.name).to eq("Node")
9     end
10  end
11 end

```

spec/models/list_spec.rb

```

1 require 'spec_helper'
2
3 describe List do
4   describe 'instantiation' do
5     let!(:list) { build(:list) }
6
7     it 'instantiates a list' do
8       expect(list.class.name).to eq("List")
9     end
10  end
11 end

```

Try running the tests `$ rspec spec`.

You should get an error saying that node and list factories have not been registered.

FF

Failures:

- 1) List instantiation instantiates a list
 Failure/Error: let!(:list) { build(:list) }
 ArgumentError:
 Factory not registered: list
 # ./spec/models/list_spec.rb:5:in `block (3 levels) in <top (required)>'
- 2) Node instantiation instantiates a list

```
Failure/Error: let!(:node) { build(:node) }  
ArgumentError:  
  Factory not registered: node  
# ./spec/models/node_spec.rb:5:in `block (3 levels) in <top (required)>'
```

Finished in 0.00074 seconds
2 examples, 2 failures

Failed examples:

```
rspec ./spec/models/list_spec.rb:7 # List instantiation instantiates a list  
rspec ./spec/models/node_spec.rb:7 # Node instantiation instantiates a list
```

The problem is caused by RSpec not being able to find FactoryGirl files.

Add FactoryGirl files as dependancies running RSpec.

spec/spec_helper.rb

```
1 require 'rubygems'  
2 require 'factory_girl'  
3  
4 require_relative '../lib/models/list'  
5 require_relative '../lib/models/node'  
6 require_relative '../spec/factories/lists.rb'  
7 require_relative '../spec/factories/nodes.rb'  
8  
9 RSpec.configure do |config|  
10   config.include FactoryGirl::Syntax::Methods  
11 end
```

Now you should be able to run tests successfully.

..

Finished in 0.00151 seconds
2 examples, 0 failures

Posted by Jason Kim Feb 4th, 2014 [ruby](#), [testing](#)

[Tweet](#)

[\[Coursera Algorithms I: Week 1\] Quick Find and Quick Union in Java »](#)

Comments

7 Comments

My Bread

 Login ▾

 Recommend

 Share

Sort by Best ▾



Join the discussion...



Christopher Mamian • 2 months ago

Great writeup! Helped a lot!!

^ | ▾ • Reply • Share >



Eric Chiang • 6 months ago

Thanks! I was scratching my head trying to figure out the meaning of `Factory not registered`. This is super helpful!

^ | ▾ • Reply • Share >

**Jason Kim** Mod → Eric Chiang • 6 months ago

Glad it was helpful.

^ | v • Reply • Share >

**Jessica Nickel** • a year ago

Thank you! This is really clear for someone who has never set up a factory using factory girl!

^ | v • Reply • Share >

**Jason Kim** Mod → Jessica Nickel • a year ago

:D Glad this could be some use!

1 ^ | v • Reply • Share >

**Andrew** • a year agoThis process is essentially the same for when I'm creating a rails application only that I'd use the 'factory_girl_rails' gem correct? <https://github.com/thoughtbot/...>

^ | v • Reply • Share >

**Jason Kim** Mod → Andrew • a year ago

Ya it should be similar more or less. Note that in rspec-rails 3.x, they started using `require "rails_helper"` instead of `require "spec_helper"`.

^ | v • Reply • Share >

ALSO ON MY BREAD

WHAT'S THIS?

[Coursera Algorithms I: Week 1] Quick Find and Quick Union in Java

11 comments • a year ago

Jose Sánchez Palacios — Hello Jason, , in the lecture when he explain the union(6,1) in the demo it results in the values connected by 1 , why not by 0 (Cero).**Simple Linked List Queue in Swift**

2 comments • a year ago

Kare Morstol — Hi, you don't need a counter or a doubly linked list for a queue. For a simpler example that supports simultaneous adding and removing, see ...**Unfortunate State of Mobile Web**

3 comments • 2 years ago

Chris Love — That is why I wrote my book, <http://amzn.to/1b0twcm>**Setting Up Bower on Rails 4 for Heroku**

1 comment • 5 months ago

Dilshod Samatov — If i understand correct command "npm install" need to run with heroku run, if so i get error npm: command not found.**Jason Kim**

A software engineer working for the music streaming company, Rhapsody in Seattle, WA.

Recent Posts

- [Setting Up Phoenix 1.0 on Mac OS X Yosemite \(With Troubleshooting Steps\)](#)
- [Don't Say Something Is Easy](#)
- [Unfair](#)
- [Setting Up Bower on Rails 4 for Heroku](#)
- [Metaprogramming in Ruby Should Be Avoided if Possible](#)

Copyright © 2015 - Jason Kim - Powered by [Octopress](#)