

Testing in Ruby on Rails

By: Haitham Mohammad



Agenda



- **Rails test directory structure**
- **Test fixtures and mocks**
- **Unit tests**
- **Functional tests**
- **Integration tests**
- **Good practices**

Directory structure



Everything has its own place to go

- fixtures
- functional
- integration
- mocks
- unit
- test_helper.rb

Test fixtures



- Fixtures are pre-set values for database records used for testing.
- Fixtures are loaded into the test database before any test is run
- Can use dynamic values using erb tags
`created_at: <%=Time.now%>`
- Easily accessible as data sets inside the test cases code

```
customers(:customer_one).name
```

- Mocks are code stubs to replace the actual implementation of a model/controller functionality at test runtime
- Useful when consuming outer dependencies

```
require 'original_class.rb'  
class OriginalClass  
  def original_method  
    #return mock_value  
  end  
end
```

Unit tests



- Unit tests are used to provide test coverage for models
- Usually involves assertions on database values or on return values of model methods

<http://ruby-doc.org/core/classes/Test/Unit/Assertions.html>

Functional tests



- Functional tests are used to provide test coverage for controllers
- Usually involves assertions on action's response headers and body

<http://api.rubyonrails.org/classes/ActionController/Assertions/ResponseAssertions.htm>

<http://api.rubyonrails.org/classes/ActionController/Assertions/DomAssertions.html>

<http://api.rubyonrails.org/classes/ActionController/Assertions/RoutingAssertions.html>

Integration tests



- Integration tests are used to test compound scenarios spanning multiple controllers and actions
- Unlike functional tests, it tests the whole stack including dispatching and routing

<http://api.rubyonrails.org/classes/ActionController/IntegrationTest.html>

How to do it



- Always use generators
- Generating a model generates its unit test stub and its fixture file
- Generating a controller generates its functional test stub
- There is a generator for integration test stub

How to do it



Before running the tests, we have to copy the development schema to the test one

```
rake db:test:prepare
```

Then run the test cases

```
rake test (all tests)
```

```
rake test:units (unit tests)
```

```
rake test:functionals (functional tests)
```

```
rake test:integration (integration tests)
```

- **Test Driven Development**
- **Disregard views for maximum coverage**
- **Ruby is Interpreted - Even trivial test cases can be useful**
- **Never mind testing private methods**
- **Never mock your own implementation - only outsiders are to be mocked**

Good Practices



- Use your test cases as a quality metric for your code - Skinny controllers, Fat models.

```
def test_next_job
  job = jobs(:transcription_job_five)
  get :show, :company_id => companies(:one).id,
      :agent_id => agents(:three).id

  # what is the problem with this line ?
  assert_nil Job.find_by_id(job.id)

  assert_response :ok
  assert_dom_equal @response.body, job.to_xml
end
```

Thank You?

e space[®]

