



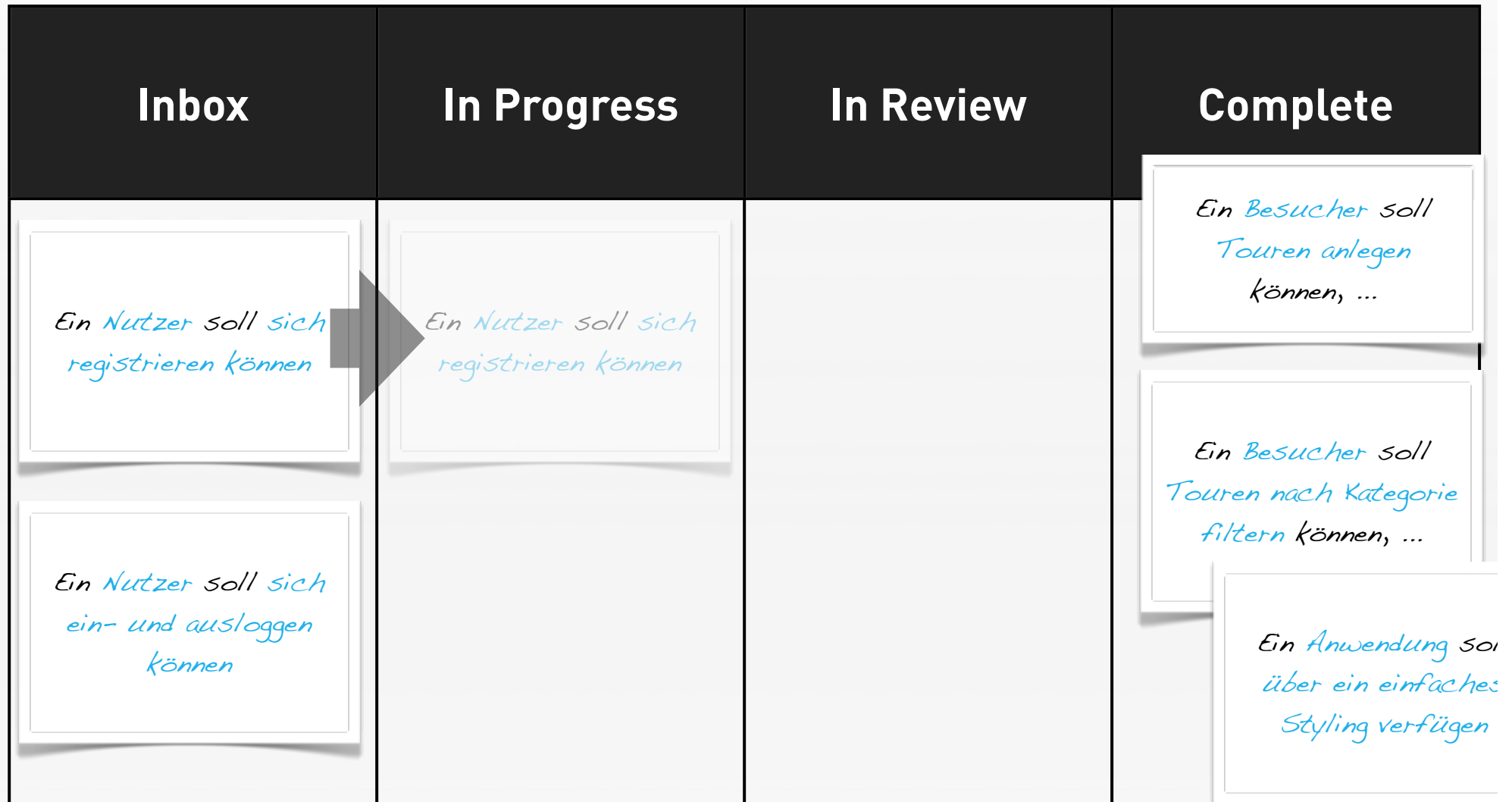
Vorlesung 9

Taskboard

RECAP

Inbox	In Progress	In Review	Complete
		<div>Ein Besucher soll Touren anlegen können, ...</div> <div>Ein Besucher soll Touren nach Kategorie filtern können, ...</div> <div>Ein Anwendung soll über ein einfaches Styling verfügen</div>	<div>Ein Besucher soll Touren anlegen können, ...</div> <div>Ein Besucher soll Touren nach Kategorie filtern können, ...</div> <div>Ein Anwendung soll über ein einfaches Styling verfügen</div>

Taskboard



User Story 4

*Ein Nutzer soll sich registrieren
können*

RUBYGEMS

- Offizielles Paketsystem für Programmiersprache Ruby
- Stellt folgendes zur Verfügung:
 - Paketformat
 - Werkzeug zur Verwaltung von Paketen
 - Repository für deren Verteilung
- Ermöglicht Verwaltung mehrerer Versionen eines Pakets
- Offizielles Verzeichnis: <http://rubygems.org/>

RUBYGEMS

(Rails/Ruby) Anwendung

Rubygems

Native Pakete

Betriebssystem

Devise

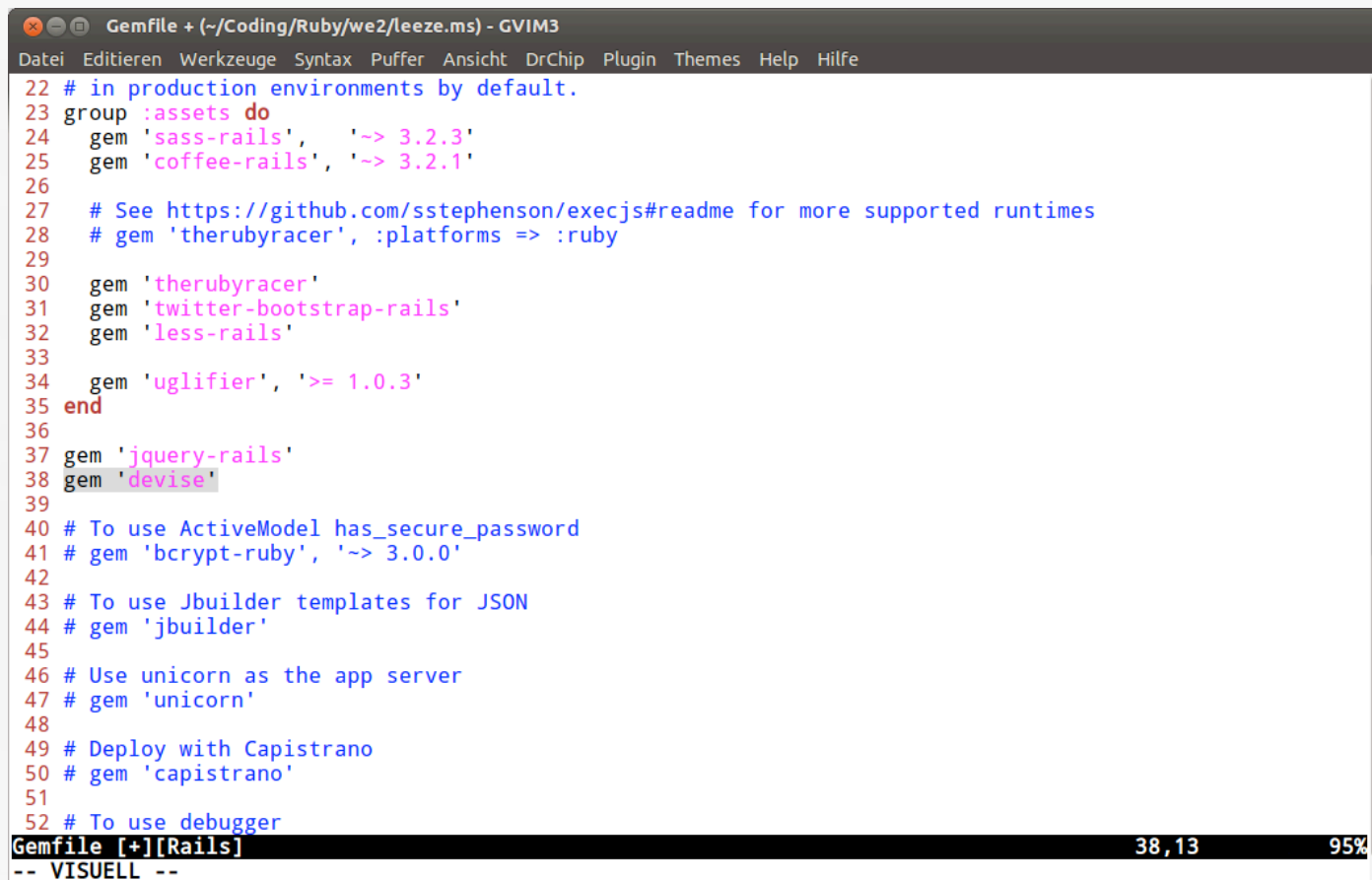
- Rubygem mit Funktionen zur Nutzerauthentifizierung
- De-facto-Standard für Rails-Applikationen
 - Sehr gute Integration in Rails
- Gem enthält eine eigene kleine Rails-Applikation
 - Controller
 - Views
 - Routen
 - ...

Devise

- Funktionen:
 - Nutzerregistrierung
 - Versand von Registrierungsbestätigungen
 - Login, Logout
 - „Passwort vergessen“-Funktion
 - Nutzer sperren
 - ...
- Funktionen sind in Modulen umgesetzt
 - Module lassen sich nach Bedarf aktivieren und deaktivieren

Devise installieren

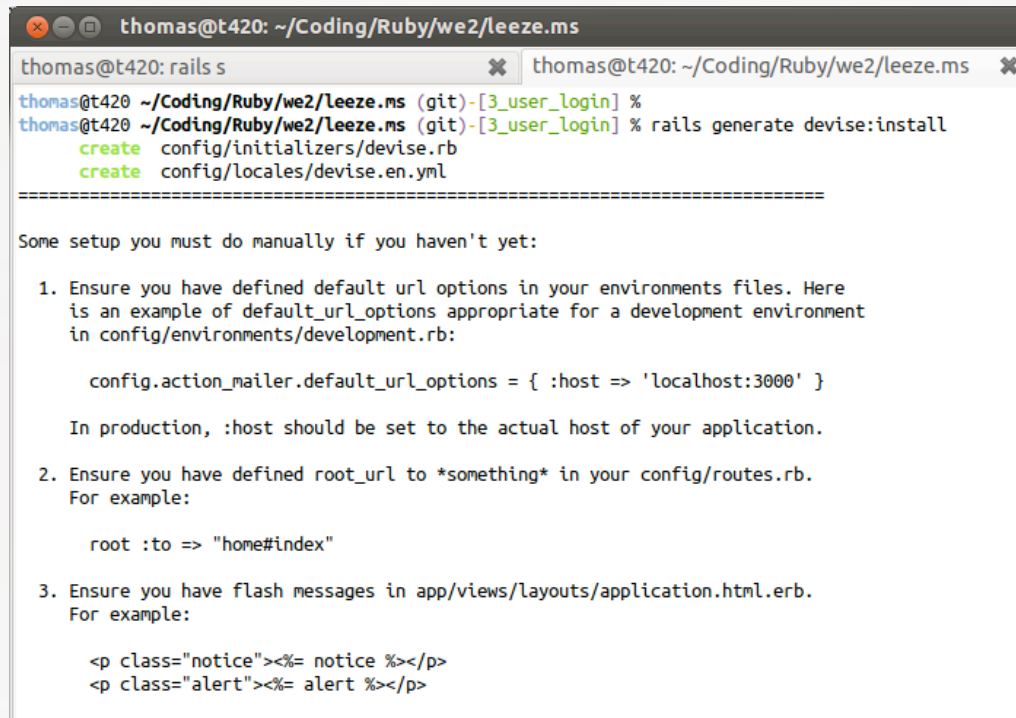
- gem 'devise' ins Gemfile aufnehmen



```
Gemfile + (~/.Coding/Ruby/we2/leeze.ms) - GVIM3
Datei Editieren Werkzeuge Syntax Puffer Ansicht DrChip Plugin Themes Help Hilfe
22 # in production environments by default.
23 group :assets do
24   gem 'sass-rails', '~> 3.2.3'
25   gem 'coffee-rails', '~> 3.2.1'
26
27   # See https://github.com/sstephenson/execjs#readme for more supported runtimes
28   # gem 'therubyracer', :platforms => :ruby
29
30   gem 'therubyracer'
31   gem 'twitter-bootstrap-rails'
32   gem 'less-rails'
33
34   gem 'uglifier', '>= 1.0.3'
35 end
36
37 gem 'jquery-rails'
38 gem 'devise'
39
40 # To use ActiveRecord has_secure_password
41 # gem 'bcrypt-ruby', '~> 3.0.0'
42
43 # To use Jbuilder templates for JSON
44 # gem 'jbuilder'
45
46 # Use unicorn as the app server
47 # gem 'unicorn'
48
49 # Deploy with Capistrano
50 # gem 'capistrano'
51
52 # To use debugger
Gemfile [~][Rails] 38,13 95%
-- VISUELL --
```

Devise installieren

- \$ bundle install
- \$ rails generate devise:install



```
thomas@t420: ~/Coding/Ruby/we2/leeze.ms
thomas@t420: rails s
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rails generate devise:install
create config/initializers/devise.rb
create config/locales/devise.en.yml

=====

Some setup you must do manually if you haven't yet:

1. Ensure you have defined default url options in your environments files. Here
   is an example of default_url_options appropriate for a development environment
   in config/environments/development.rb:

       config.action_mailer.default_url_options = { :host => 'localhost:3000' }

   In production, :host should be set to the actual host of your application.

2. Ensure you have defined root_url to *something* in your config/routes.rb.
   For example:

       root :to => "home#index"

3. Ensure you have flash messages in app/views/layouts/application.html.erb.
   For example:

       <p class="notice"><%= notice %></p>
       <p class="alert"><%= alert %></p>
```

Devise installieren

- config/environments/development.rb anpassen

A screenshot of a code editor window with a dark theme. The title bar at the top reads "config/environments/development.rb". The editor area shows a few lines of Ruby code: a comment "# ...", a line "config.action_mailer.default_url_options = { :host => 'localhost:3000' }", and another comment "# ...".

```
config/environments/development.rb

# ...
config.action_mailer.default_url_options = { :host => 'localhost:3000' }
# ...
```

Devise einrichten

- Devise speichert Nutzer als ActiveRecord-Model
- Erzeugtes Model ist konfigurierbar
- Erstellung eines Devise User-Models:
\$ rails generate devise User
- Ausführen der erzeugten Migration:
\$ rake db:migrate

Devise einrichten



```
thomas@t420: ~/Coding/Ruby/we2/leeze.ms
thomas@t420: rails s
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rails generate devise User
  invoke  active_record
  create  db/migrate/20121213162627_devise_create_users.rb
  create  app/models/user.rb
  invoke  rspec
  create  spec/models/user_spec.rb
  invoke  factory_girl
  create  spec/factories/users.rb
  insert  app/models/user.rb
  route  devise_for :users
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %
```

Devise einrichten

- Devise enthält eine Rails-Engine
- Devise-Controller, -Routen und -Views sind in eigener Applikation verfügbar
- Funktionalität muss nicht neu implementiert werden

Devise einrichten

- \$ rake routes

```
thomas@t420: ~/Coding/Ruby/we2/leeze.ms
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rake routes
      new_user_session GET    /users/sign_in(.:format)  devise/sessions#new
      user_session     POST   /users/sign_in(.:format)  devise/sessions#create
destroy_user_session   DELETE /users/sign_out(.:format) devise/sessions#destroy
      user_password    POST   /users/password(.:format) devise/passwords#create
      new_user_password GET     /users/password/new(.:format) devise/passwords#new
      edit_user_password GET     /users/password/edit(.:format) devise/passwords#edit
                        PUT     /users/password(.:format)  devise/passwords#update
cancel_user_registration GET     /users/cancel(.:format)   devise/registrations#cancel
      user_registration POST    /users(.:format)          devise/registrations#create
      new_user_registration GET     /users/sign_up(.:format)  devise/registrations#new
      edit_user_registration GET     /users/edit(.:format)     devise/registrations#edit
                        PUT     /users(.:format)          devise/registrations#update
                        DELETE  /users(.:format)          devise/registrations#destroy
      tours            GET     /tours(.:format)          tours#index
                        POST    /tours(.:format)          tours#create
      new_tour         GET     /tours/new(.:format)      tours#new
      edit_tour        GET     /tours/:id/edit(.:format) tours#edit
      tour             GET     /tours/:id(.:format)      tours#show
                        PUT     /tours/:id(.:format)      tours#update
                        DELETE  /tours/:id(.:format)      tours#destroy
      category_tours   GET     /:category_id(.:format)   tours#index
      root             /                               tours#index
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %
```

Story 4: Nutzerregistrierung

- Unter `http://localhost:3000/users/sign_up` stellt Devise ein Registrierungsformular bereit
- Registrierung funktioniert bereits!
- Daher: Integration Test für dieses Formular schreiben

Story 4: Nutzerregistrierung

The screenshot shows a web browser window with the address bar at `localhost:3000/users/sign_up`. The page title is 'LeezeMs'. The browser's address bar shows several tabs: 'Abonnieren ...', 'JS-Test', '(864e5));docum...', 'performance - I...', 'little-scale: BAS...', and 'arsduo/koala - ...'. The page has a navigation bar with links: 'LeezeMs', 'Radtour', 'Mountainbiketour', 'Radwandern', 'My Category', and 'My Category'. The main content area is titled 'Sign up' and contains three input fields: 'Email', 'Password', and 'Password confirmation'. Below these fields is a 'Sign up' button. At the bottom of the form, there are links for 'Sign in' and 'Forgot your password?'. A sidebar on the left is titled 'SIDEBAR' and contains links 'Link1', 'Link2', and 'Link3'. The footer of the page reads '© Company 2012'.

LeezeMs

Radtour Mountainbiketour Radwandern My Category My Category

SIDEBAR

Link1
Link2
Link3

Sign up

Email

Password

Password confirmation

Sign up

Sign in
Forgot your password?

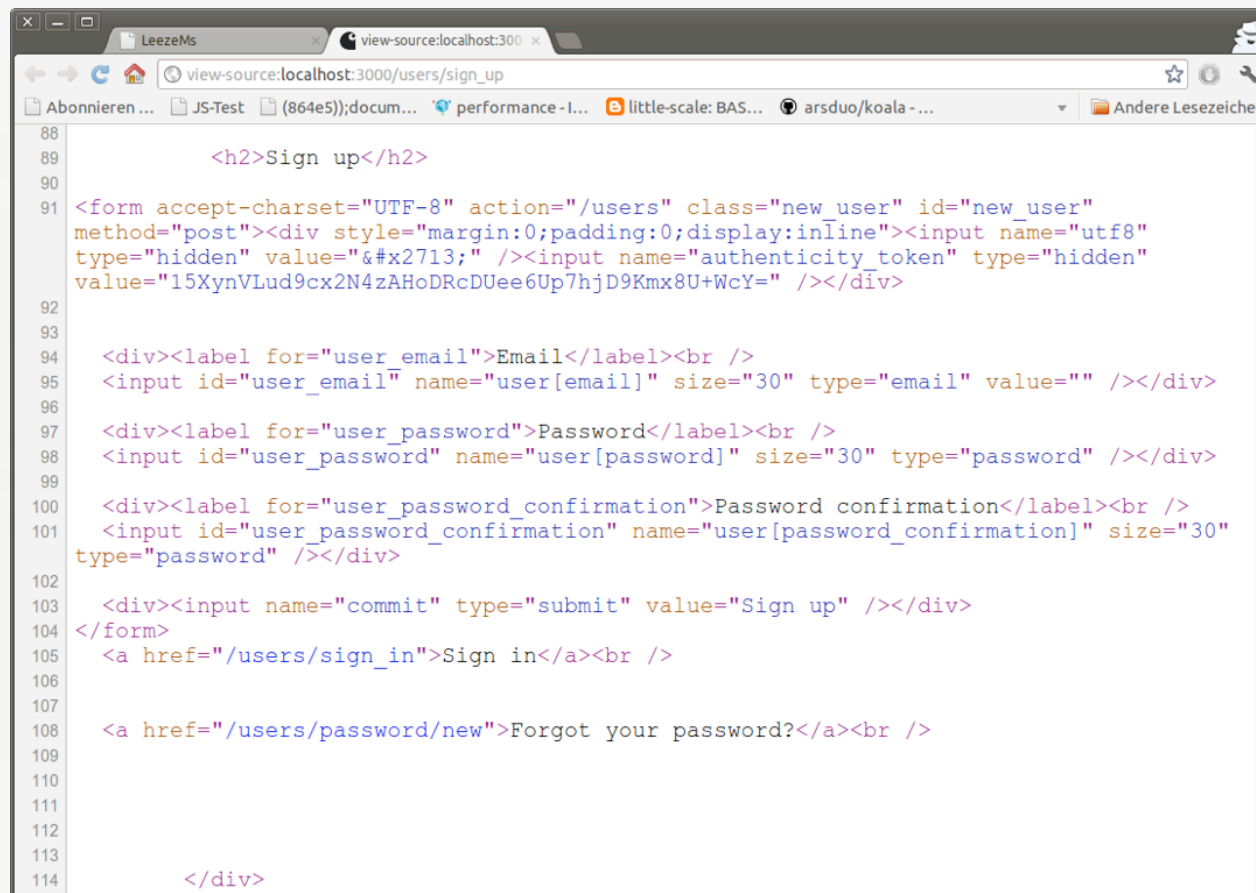
© Company 2012

Story 4: Integration Test

- Der Integration Test könnte so ablaufen:
 - Klicke auf Link „Registrieren“
 - Fülle Formular aus
 - Klicke auf Button „Registrieren“
 - Es sollte ein Nutzer erstellt werden
 - Nutzer sollte eingeloggt sein
- Hierfür brauchen wir die Namen der Formularfelder und der Buttons

Story 4: Integration Test

- Quelltext der Registrierungsseite:



```
88      <h2>Sign up</h2>
89
90      <form accept-charset="UTF-8" action="/users" class="new_user" id="new_user"
91      method="post"><div style="margin:0;padding:0;display:inline"><input name="utf8"
92      type="hidden" value="&#x2713;" /><input name="authenticity_token" type="hidden"
93      value="15XynVLud9cx2N4zAHoDRcDUee6Up7hjD9Kmx8U+WcY=" /></div>
94
95      <div><label for="user_email">Email</label><br />
96      <input id="user_email" name="user[email]" size="30" type="email" value="" /></div>
97
98      <div><label for="user_password">Password</label><br />
99      <input id="user_password" name="user[password]" size="30" type="password" /></div>
100
101      <div><label for="user_password_confirmation">Password confirmation</label><br />
102      <input id="user_password_confirmation" name="user[password_confirmation]" size="30"
103      type="password" /></div>
104
105      <div><input name="commit" type="submit" value="Sign up" /></div>
106    </form>
107    <a href="/users/sign_in">Sign in</a><br />
108
109    <a href="/users/password/new">Forgot your password?</a><br />
110
111
112
113
114  </div>
```

Story 4: Integration Test

- Resultierender Test:

```
spec/features/login_spec.rb

require 'spec_helper'

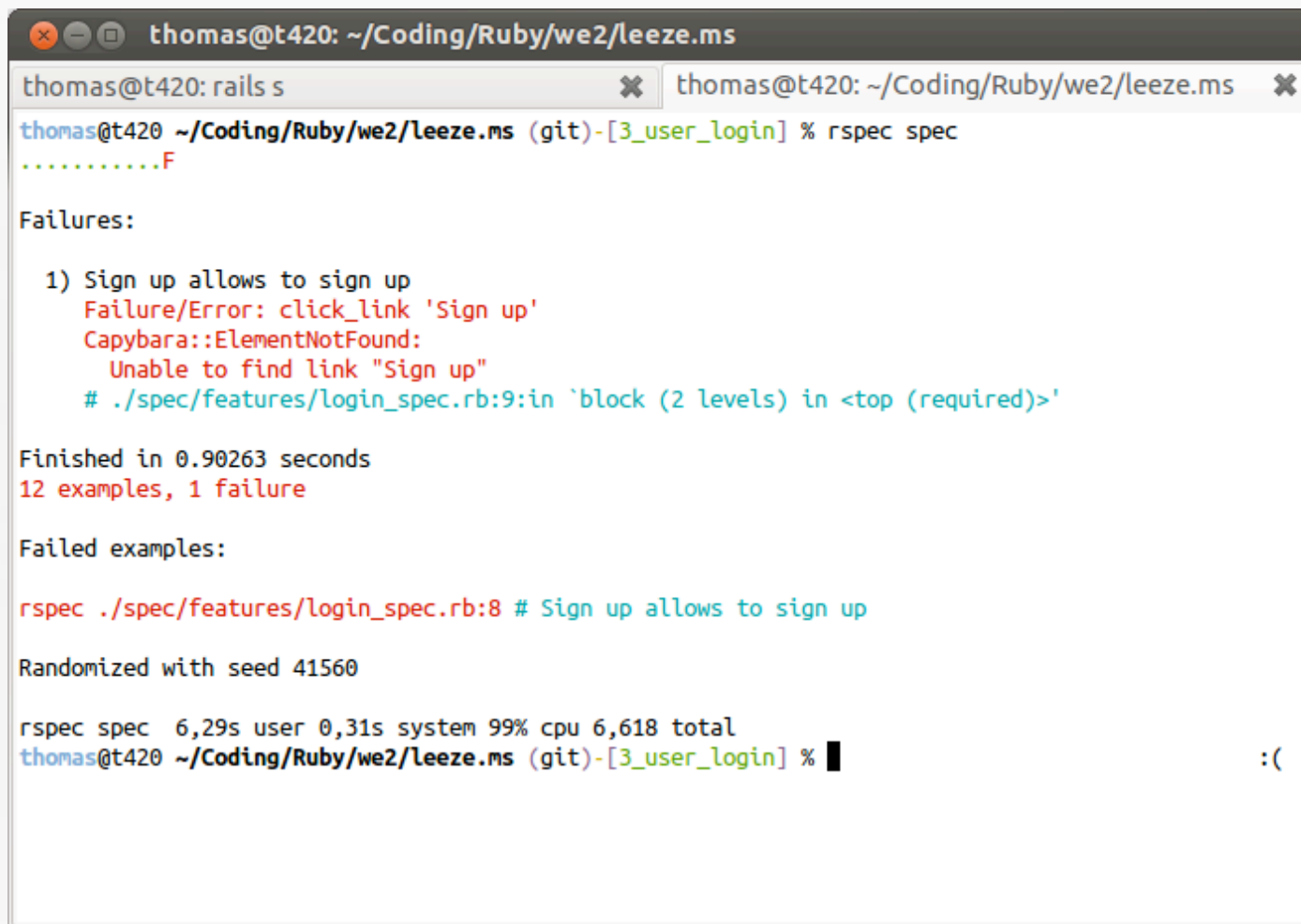
describe 'Sign up' do
  before(:each) do
    visit root_path
  end

  it 'allows to sign up' do
    click_link 'Sign up'
    fill_in 'user_email', with: 'thomas.hollstegge@zweitag.de'
    fill_in 'user_password', with: 'monkey123'
    fill_in 'user_password_confirmation', with: 'monkey123'

    expect { click_button 'Sign up' }.to change { User.count }.by(1)

    page.should have_content 'Welcome, thomas.hollstegge@zweitag.de'
  end
end
```

Story 4: Integration Test



A terminal window titled 'thomas@t420: ~/Coding/Ruby/we2/leeze.ms' showing the execution of an integration test. The user runs 'rspec spec' and the output shows 12 examples with 1 failure. The failure is in the 'Sign up allows to sign up' example, where the 'click_link' method failed because it could not find the 'Sign up' link. The error message is 'Capybara::ElementNotFound: Unable to find link "Sign up"'. The terminal also shows the test finished in 0.90263 seconds and was randomized with seed 41560.

```
thomas@t420: ~/Coding/Ruby/we2/leeze.ms
thomas@t420: rails s
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rspec spec
.....F

Failures:

  1) Sign up allows to sign up
     Failure/Error: click_link 'Sign up'
     Capybara::ElementNotFound:
       Unable to find link "Sign up"
     # ./spec/features/login_spec.rb:9:in `block (2 levels) in <top (required)>'

Finished in 0.90263 seconds
12 examples, 1 failure

Failed examples:

rspec ./spec/features/login_spec.rb:8 # Sign up allows to sign up

Randomized with seed 41560

rspec spec 6,29s user 0,31s system 99% cpu 6,618 total
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % █
```

Story 4: Integration Test

- Schritt 1: Navigation in ein Partial auslagern

```
app/views/layouts/application.html.erb

<!-- ... -->
<body>

  <%= render 'shared/navigation' %>

  <div class="container-fluid">
    <div class="row-fluid">
      <div class="span12">
        <%= bootstrap_flash %>
        <%= yield %>
      </div>
    </div><!--/row-->
    <footer>
      <p>&copy; leeze.ms 2013</p>
    </footer>

  </div> <!-- /container -->

</body>
<!-- ... -->
```

Story 4: Integration Test

● Schritt 2: Registrierungs-Link einbauen

```
app/views/shared/_navigation.html.erb

<div class="navbar navbar-fluid-top">
  <div class="navbar-inner">
    <div class="container-fluid">
      <a class="btn btn-navbar" data-target=".nav-collapse" data-toggle="collapse">
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </a>
      <a class="brand" href="#">LeezeMs</a>
      <div class="container-fluid nav-collapse">
        <ul class="nav">
          <% Category.all.each do |category| %>
            <li><%= link_to category.name, category_tours_path(category) %></li>
          <% end %>
          <li><%= link_to 'Sign up', new_user_registration_path %></li>
        </ul>
      </div><!--/.nav-collapse -->
    </div>
  </div>
</div>
```

Story 4: Integration Test

- Tests ausführen: `$ rspec spec`

```
[RSpec results] 1 examples, 1 failures
Failures:

  1) Sign up allows to sign up
     Failure/Error: page.should have_content 'Welcome, thomas.hollstegge@zweitag.de'
           expected there to be text "Welcome, thomas.hollstegge@zweitag.de" in "Lee
zeMs Sign up Sidebar Link1 Link2 Link3 xWelcome! You have signed up successfully
. Title New Tour © Company 2012"
     # ./spec/features/login_spec.rb:16:in `block (2 levels) in <top (required)>'

Finished in 0.67025 seconds
1 example, 1 failure

Failed examples:

rspec ./spec/features/login_spec.rb:8 # Sign up allows to sign up
16:51:05 - ERROR - Error sending notification with libnotify: cannot load such f
ile -- libnotify

Randomized with seed 7173

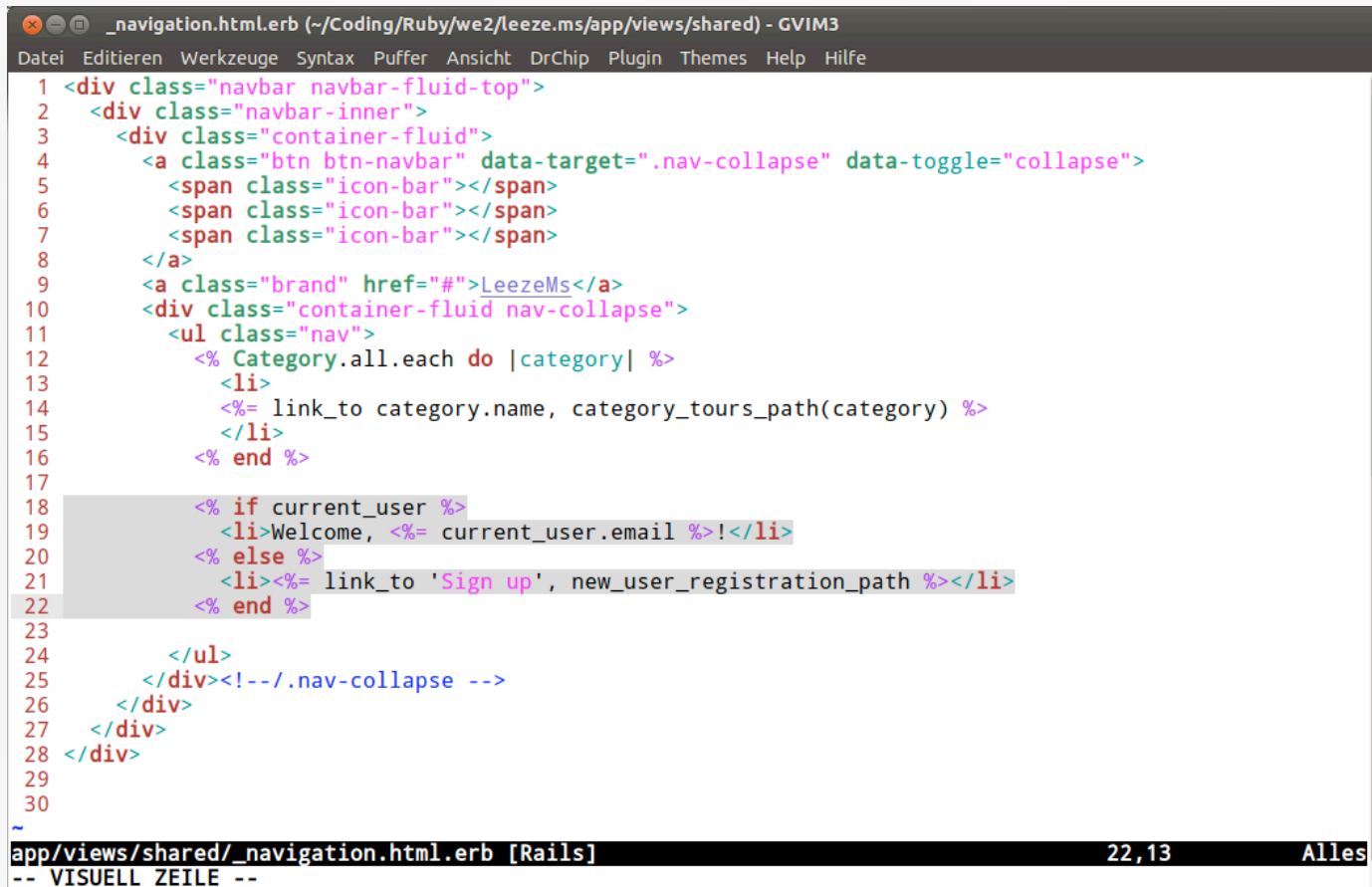
[1] guard(main)> █
```


Story 4: Integration Test

- Devise stellt eine `current_user` Methode bereit
- Nutzer eingeloggt \Leftrightarrow `current_user` enthält User-Instanz
- Nutzer nicht eingeloggt \Leftrightarrow `current_user == nil`
- Daher: Falls `current_user` gesetzt ist, zeige die Mailadresse an. Falls nicht, zeige einen Registrierungs-Link an

Story 4: Integration Test

- Schritt 3: Mailadresse des Nutzers anzeigen

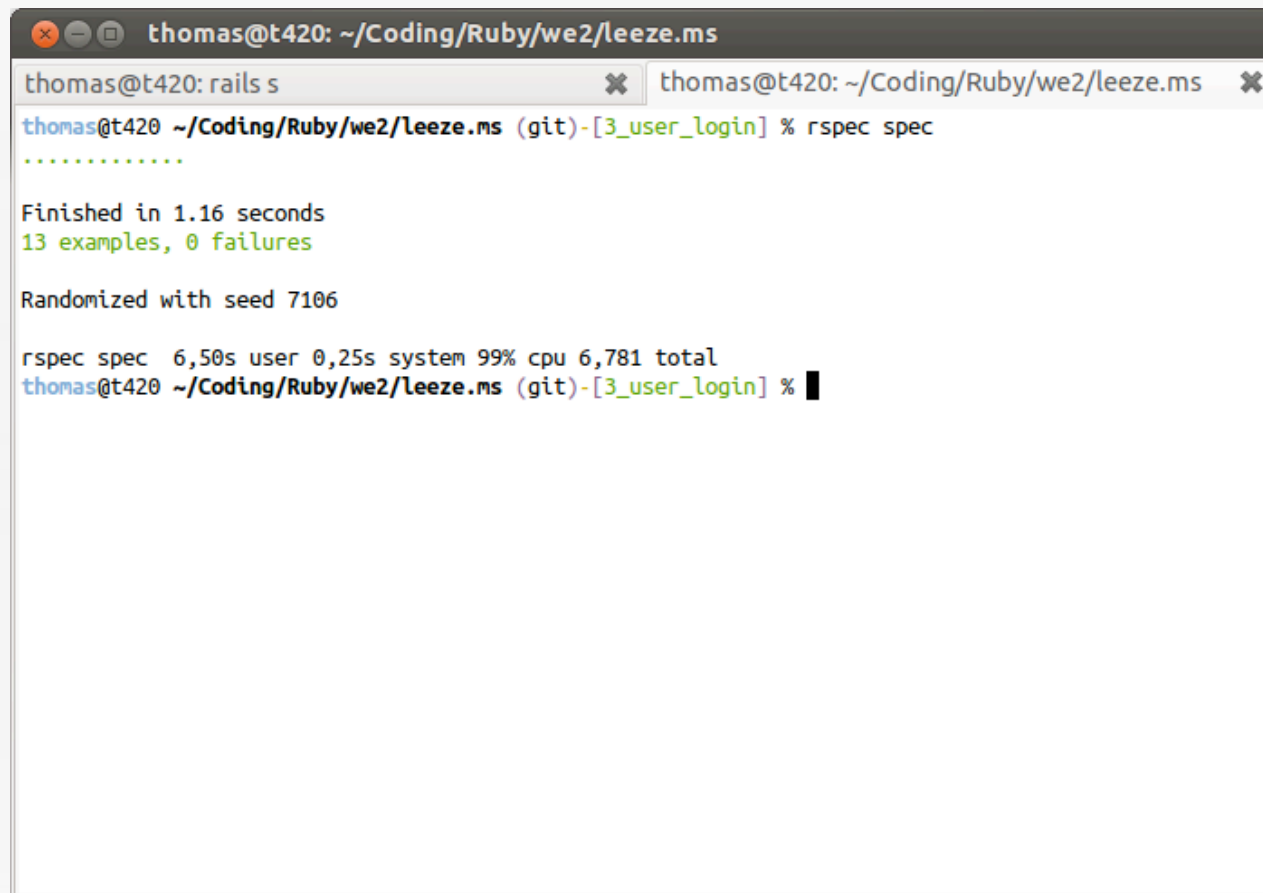


```
_navigation.html.erb (~/.Coding/Ruby/we2/leeze.ms/app/views/shared) - GVIM3
Datei Editieren Werkzeuge Syntax Puffer Ansicht DrChip Plugin Themes Help Hilfe

1 <div class="navbar navbar-fluid-top">
2   <div class="navbar-inner">
3     <div class="container-fluid">
4       <a class="btn btn-navbar" data-target=".nav-collapse" data-toggle="collapse">
5         <span class="icon-bar"></span>
6         <span class="icon-bar"></span>
7         <span class="icon-bar"></span>
8       </a>
9       <a class="brand" href="#">LeezeMs</a>
10      <div class="container-fluid nav-collapse">
11        <ul class="nav">
12          <% Category.all.each do |category| %>
13            <li>
14              <%= link_to category.name, category_tours_path(category) %>
15            </li>
16          <% end %>
17
18          <% if current_user %>
19            <li>Welcome, <%= current_user.email %>!</li>
20          <% else %>
21            <li><%= link_to 'Sign up', new_user_registration_path %></li>
22          <% end %>
23        </ul>
24      </div><!--/.nav-collapse -->
25    </div>
26  </div>
27</div>
28</div>
29
30
~
app/views/shared/_navigation.html.erb [Rails] 22,13 Alles
-- VISUELL ZEILE --
```

Story 4: Integration Test

- Tests ausführen:



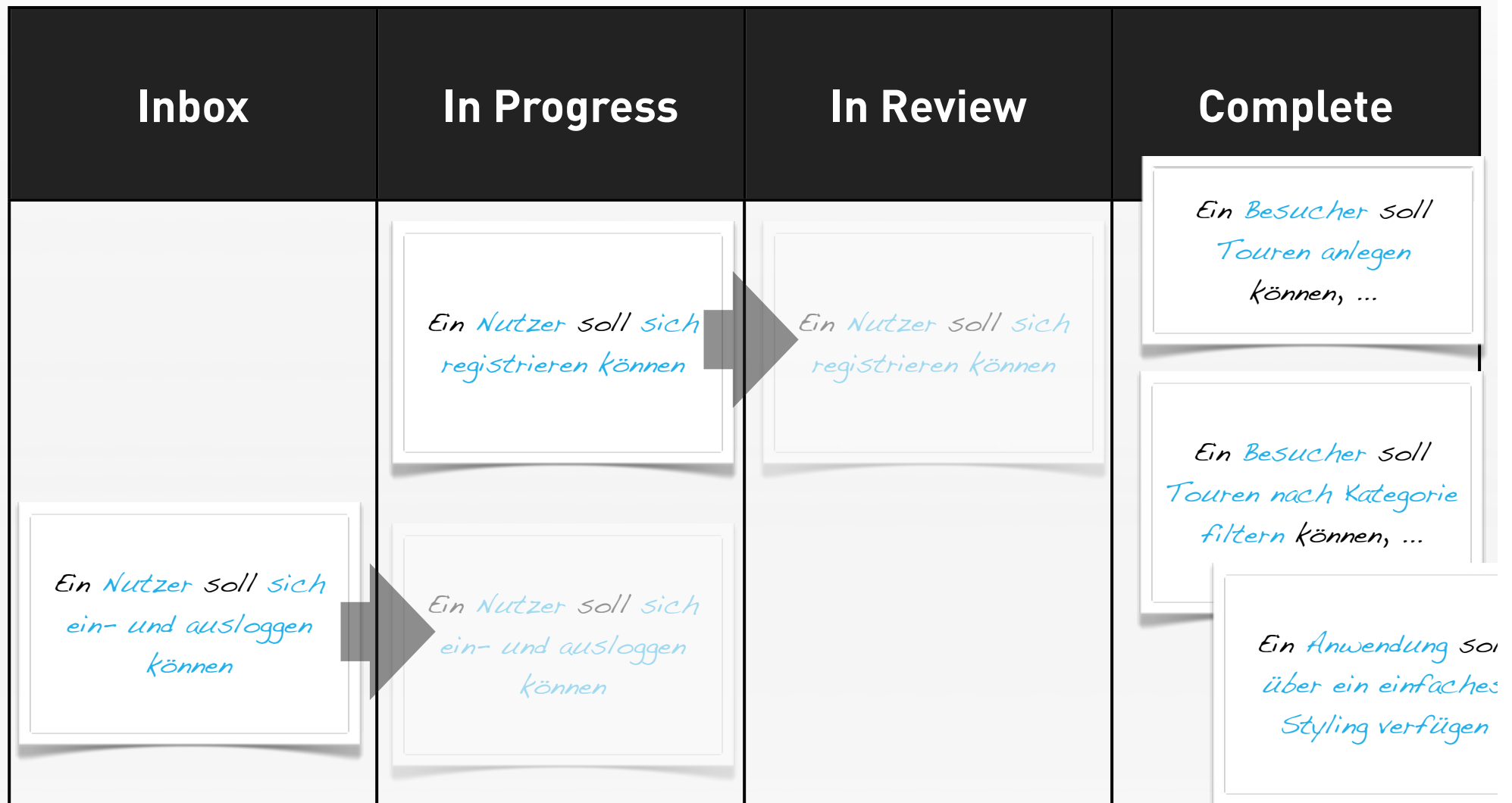
```
thomas@t420: ~/Coding/Ruby/we2/leeze.ms
thomas@t420: rails s
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rspec spec
.....

Finished in 1.16 seconds
13 examples, 0 failures

Randomized with seed 7106

rspec spec 6,50s user 0,25s system 99% cpu 6,781 total
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %
```

Taskboard



User Story 5

Ein Nutzer soll sich ein- und ausloggen können.

Integration-Test:

Ein Nutzer soll sich einloggen können.

Story 5: Einloggen

- Idee für einen Integration Test:
 - Erstelle einen Nutzer in der Datenbank
 - Klicke auf Link „Einloggen“
 - Gebe Nutzerdaten ein
 - Klicke auf Button „Einloggen“
 - Nutzer sollte eingeloggt sein
- Zur Erstellung von Test-Nutzern brauchen wir eine Factory
 - --> Hierfür nutzen wir `factory_girl`

Story 5: Einloggen

- Schritt 1: User-Factory anlegen:



A code editor window with a dark background and a title bar. The title bar contains three small circles on the left and the text 'spec/factories/users.rb' on the right. The code is written in a light gray font and defines a FactoryGirl factory for users.

```
FactoryGirl.define do
  factory :user do
    email { Faker::Internet.email }
    password 'monkey123'
    password_confirmation 'monkey123'
  end
end
```

Story 5: Einloggen

● Schritt 2: Test schreiben

```
spec/features/login_spec.rb

require 'spec_helper'

describe 'Sign up' do

  # ...

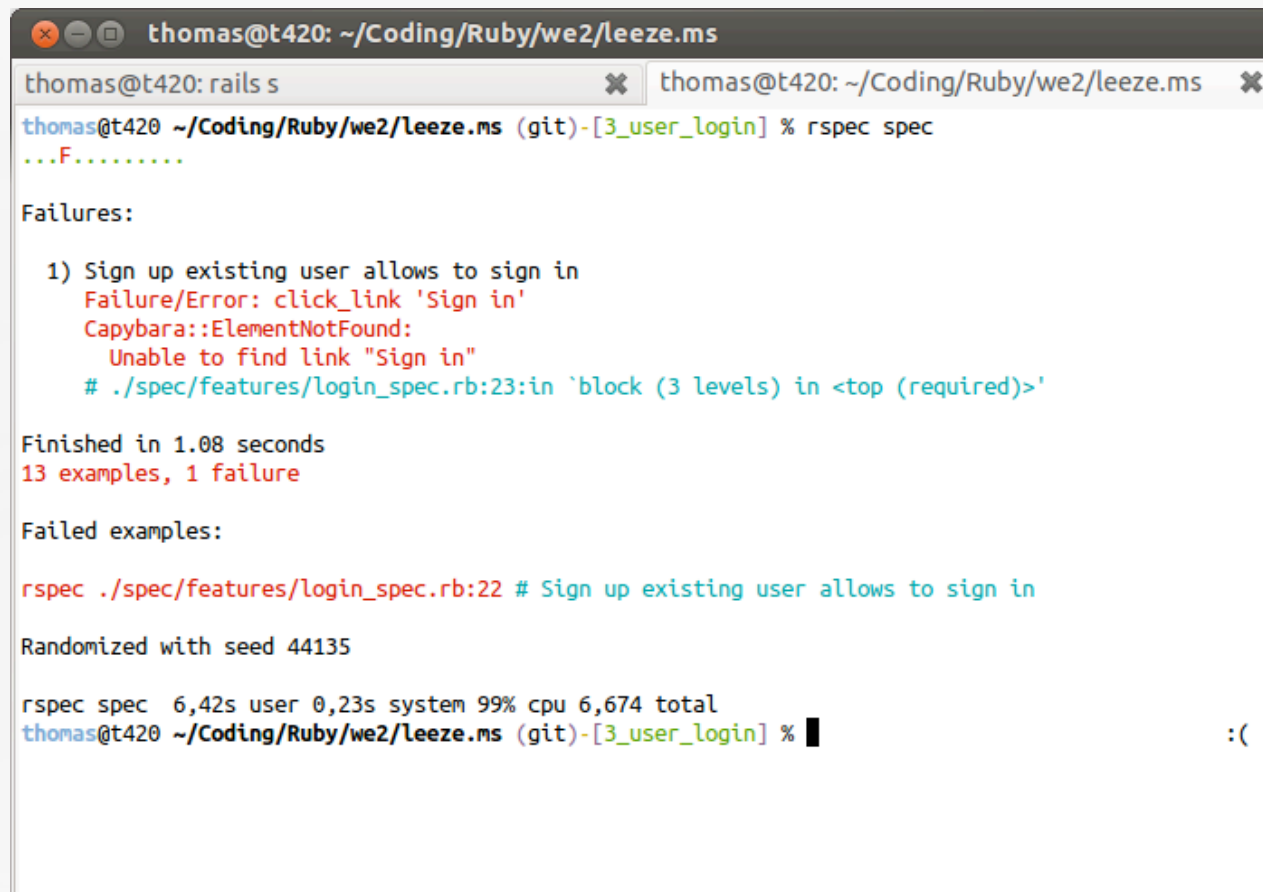
  context 'existing user' do
    let!(:user) { FactoryGirl.create :user }

    it 'allows to sign in' do
      click_link 'Sign in'
      fill_in "user_email", with: user.email
      fill_in "user_password", with: 'monkey123'
      click_button 'Sign in'

      page.should have_content "Welcome, #{user.email}"
    end
  end
end
```


Story 5: Einloggen

- Tests ausführen:



```
thomas@t420: ~/Coding/Ruby/we2/leeze.ms
thomas@t420: rails s
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rspec spec
...F.....

Failures:

  1) Sign up existing user allows to sign in
     Failure/Error: click_link 'Sign in'
     Capybara::ElementNotFound:
       Unable to find link "Sign in"
     # ./spec/features/login_spec.rb:23:in `block (3 levels) in <top (required)>'

Finished in 1.08 seconds
13 examples, 1 failure

Failed examples:

rspec ./spec/features/login_spec.rb:22 # Sign up existing user allows to sign in

Randomized with seed 44135

rspec spec 6.42s user 0.23s system 99% cpu 6.674 total
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %
```

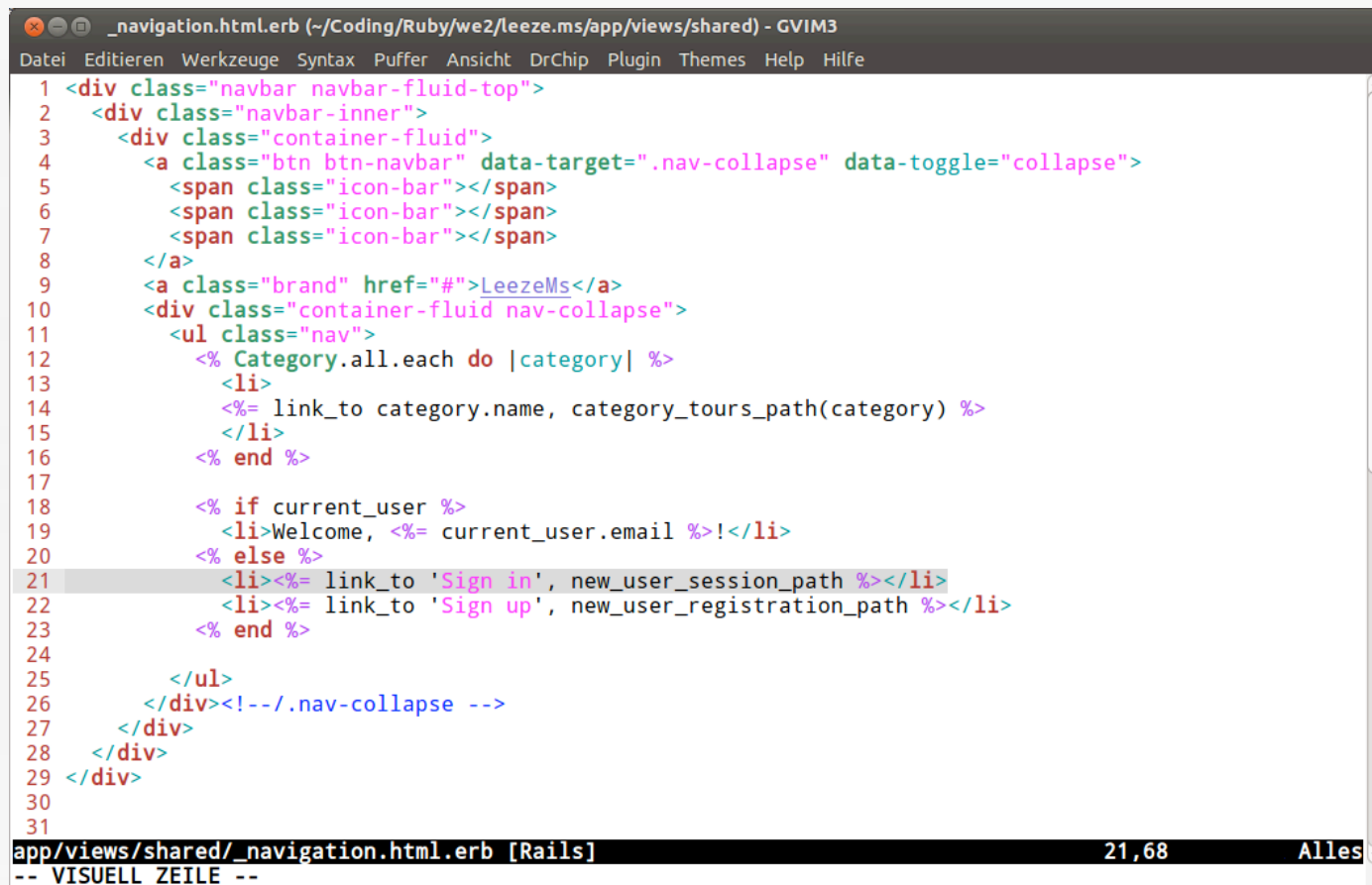
Devise einrichten

- \$ rake routes

```
thomas@t420: ~/Coding/Ruby/we2/leeze.ms
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rake routes
      new_user_session GET    /users/sign_in(.:format)  devise/sessions#new
      user_session     POST   /users/sign_in(.:format)  devise/sessions#create
destroy_user_session  DELETE /users/sign_out(.:format)  devise/sessions#destroy
      user_password    POST   /users/password(.:format)  devise/passwords#create
      new_user_password GET    /users/password/new(.:format) devise/passwords#new
      edit_user_password GET    /users/password/edit(.:format) devise/passwords#edit
                        PUT    /users/password(.:format)  devise/passwords#update
cancel_user_registration GET    /users/cancel(.:format)    devise/registrations#cancel
      user_registration POST   /users(.:format)           devise/registrations#create
      new_user_registration GET    /users/sign_up(.:format)    devise/registrations#new
      edit_user_registration GET    /users/edit(.:format)       devise/registrations#edit
                        PUT    /users(.:format)           devise/registrations#update
                        DELETE /users(.:format)           devise/registrations#destroy
      tours            GET    /tours(.:format)           tours#index
                        POST   /tours(.:format)           tours#create
      new_tour         GET    /tours/new(.:format)       tours#new
      edit_tour        GET    /tours/:id/edit(.:format)   tours#edit
      tour            GET    /tours/:id(.:format)       tours#show
                        PUT    /tours/:id(.:format)       tours#update
                        DELETE /tours/:id(.:format)       tours#destroy
      category_tours  GET    /:category_id(.:format)     tours#index
      root            /                                     tours#index
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %
```

Story 5: Einloggen

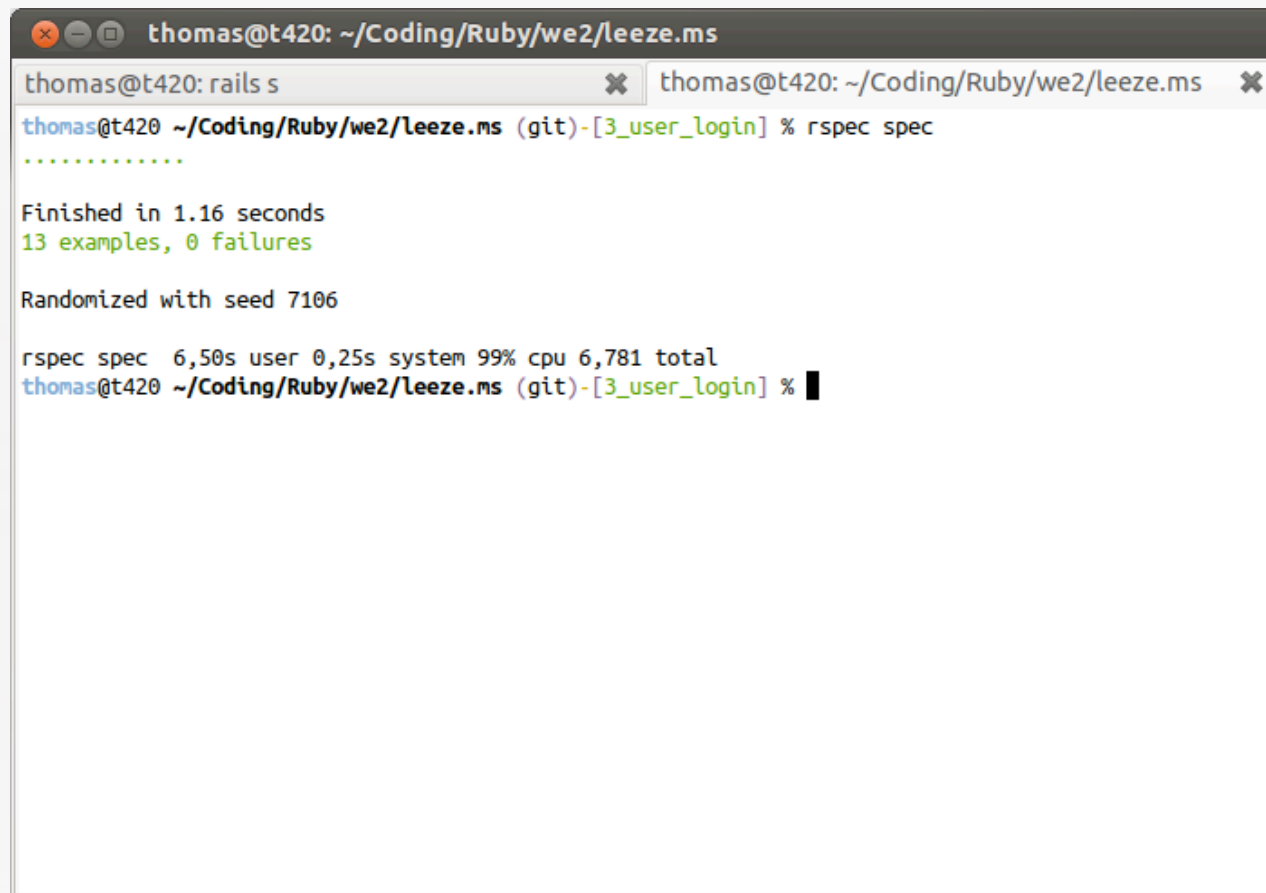
- Login-Link in Navigation einbauen



```
_navigation.html.erb (~/.Coding/Ruby/we2/leeze.ms/app/views/shared) - GVIM3
Datei Editieren Werkzeuge Syntax Puffer Ansicht DrChip Plugin Themes Help Hilfe
1 <div class="navbar navbar-fluid-top">
2   <div class="navbar-inner">
3     <div class="container-fluid">
4       <a class="btn btn-navbar" data-target=".nav-collapse" data-toggle="collapse">
5         <span class="icon-bar"></span>
6         <span class="icon-bar"></span>
7         <span class="icon-bar"></span>
8       </a>
9       <a class="brand" href="#">LeezeMs</a>
10      <div class="container-fluid nav-collapse">
11        <ul class="nav">
12          <% Category.all.each do |category| %>
13            <li>
14              <%= link_to category.name, category_tours_path(category) %>
15            </li>
16          <% end %>
17
18          <% if current_user %>
19            <li>Welcome, <%= current_user.email %>!</li>
20          <% else %>
21            <li><%= link_to 'Sign in', new_user_session_path %></li>
22            <li><%= link_to 'Sign up', new_user_registration_path %></li>
23          <% end %>
24        </ul>
25      </div><!--/.nav-collapse -->
26    </div>
27  </div>
28</div>
29</div>
30
31
app/views/shared/_navigation.html.erb [Rails] 21,68 Alles
-- VISUELL ZEILE --
```

Story 5: Einloggen

- Tests ausführen

A terminal window titled 'thomas@t420: ~/Coding/Ruby/we2/leeze.ms' showing the execution of 'rspec spec'. The output indicates that the tests passed successfully. The window has a dark title bar and a light-colored terminal area with syntax-highlighted text.

```
thomas@t420: ~/Coding/Ruby/we2/leeze.ms
thomas@t420: rails s
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rspec spec
.....

Finished in 1.16 seconds
13 examples, 0 failures

Randomized with seed 7106

rspec spec 6,50s user 0,25s system 99% cpu 6,781 total
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %
```

User Story 5

Ein Nutzer soll sich ein- und ausloggen können.

Integration-Test:

Ein Nutzer soll sich ausloggen können.

Story 5: Ausloggen

- Test schreiben

```
spec/features/login_spec.rb

require 'spec_helper'

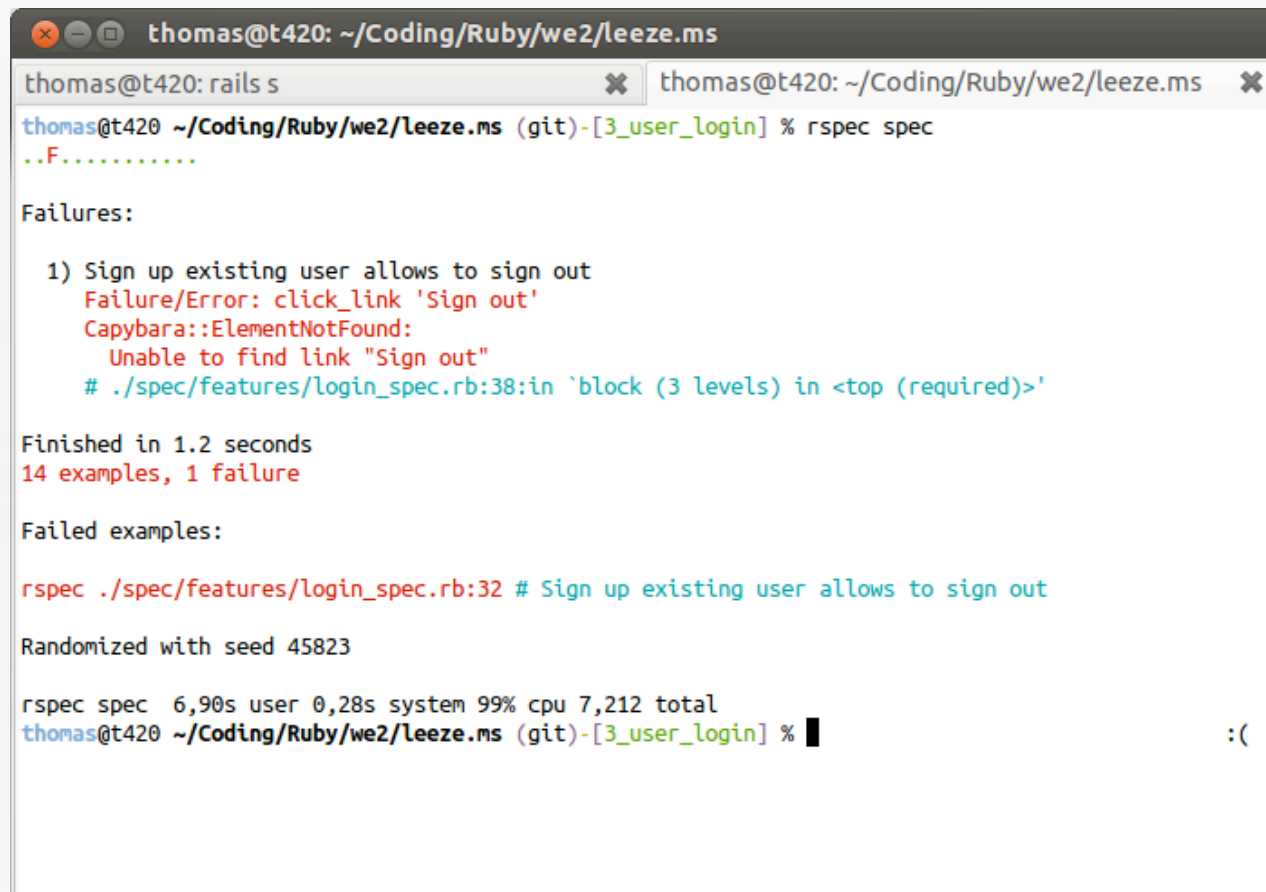
describe 'Sign up' do
  # ...
  context 'existing user' do
    let!(:user) { FactoryGirl.create :user }
    # ...
    it 'allows to sign out' do
      click_link 'Sign in'
      fill_in "user_email", with: user.email
      fill_in "user_password", with: 'monkey123'
      click_button 'Sign in'

      click_link 'Sign out'

      page.should_not have_content "Welcome, #{user.email}"
      page.should have_content 'Sign in'
    end
  end
end
```

Story 5: Ausloggen

- Tests ausführen



```
thomas@t420: ~/Coding/Ruby/we2/leeze.ms
thomas@t420: rails s
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rspec spec
..F.....

Failures:

  1) Sign up existing user allows to sign out
     Failure/Error: click_link 'Sign out'
     Capybara::ElementNotFound:
       Unable to find link "Sign out"
     # ./spec/features/login_spec.rb:38:in `block (3 levels) in <top (required)>'

Finished in 1.2 seconds
14 examples, 1 failure

Failed examples:

rspec ./spec/features/login_spec.rb:32 # Sign up existing user allows to sign out

Randomized with seed 45823

rspec spec 6,90s user 0,28s system 99% cpu 7,212 total
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %
```

Devise einrichten

- \$ rake routes

```
thomas@t420: ~/Coding/Ruby/we2/leeze.ms
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rake routes
      new_user_session GET    /users/sign_in(.:format)  devise/sessions#new
      user_session     POST   /users/sign_in(.:format)  devise/sessions#create
destroy_user_session  DELETE /users/sign_out(.:format) devise/sessions#destroy
      user_password    POST   /users/password(.:format) devise/passwords#create
      new_user_password GET     /users/password/new(.:format) devise/passwords#new
      edit_user_password GET     /users/password/edit(.:format) devise/passwords#edit
                        PUT     /users/password(.:format)  devise/passwords#update
cancel_user_registration GET    /users/cancel(.:format)  devise/registrations#cancel
      user_registration POST   /users(.:format)         devise/registrations#create
      new_user_registration GET    /users/sign_up(.:format)  devise/registrations#new
      edit_user_registration GET    /users/edit(.:format)     devise/registrations#edit
                        PUT     /users(.:format)         devise/registrations#update
                        DELETE  /users(.:format)         devise/registrations#destroy
      tours            GET    /tours(.:format)         tours#index
                        POST    /tours(.:format)         tours#create
      new_tour         GET    /tours/new(.:format)      tours#new
      edit_tour        GET    /tours/:id/edit(.:format) tours#edit
      tour            GET    /tours/:id(.:format)      tours#show
                        PUT     /tours/:id(.:format)      tours#update
                        DELETE  /tours/:id(.:format)      tours#destroy
      category_tours   GET    /:category_id(.:format)   tours#index
      root            /                                     tours#index
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %
```


Story 5: Ausloggen

- Logout-Link in Navigation einbauen

```
app/views/shared/_navigation.html.erb

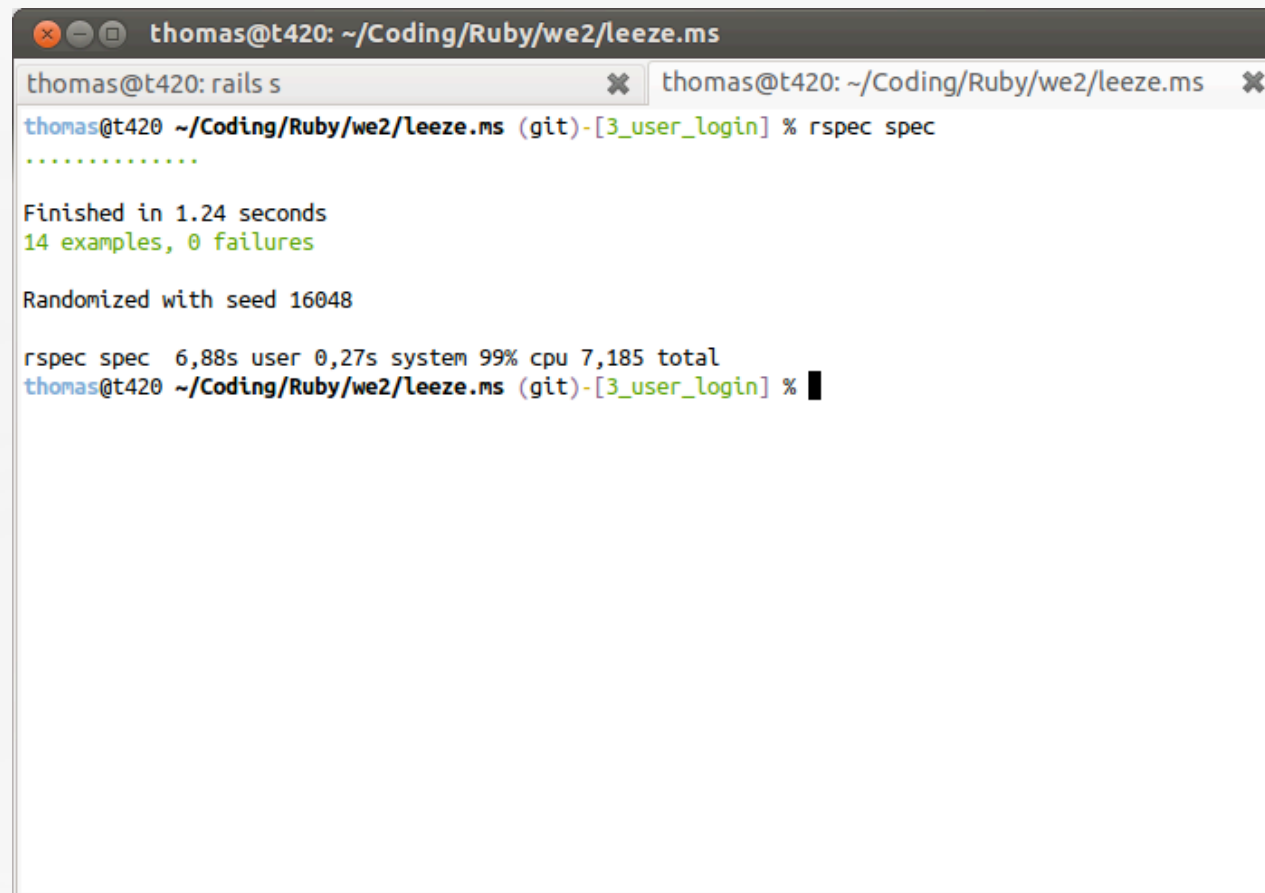
# ...

<% if current_user %>
  <li>Welcome, <%= current_user.email %>!</li>
  <li><%= link_to 'Sign out', destroy_user_session_path, method: :delete %></li>
<% else %>
  <li><%= link_to 'Sign in', new_user_session_path %></li>
  <li><%= link_to 'Sign up', new_user_registration_path %></li>
<% end %>

# ...
```

Story 5: Ausloggen

- Test ausführen

A terminal window titled 'thomas@t420: ~/Coding/Ruby/we2/leeze.ms' with two tabs: 'thomas@t420: rails s' and 'thomas@t420: ~/Coding/Ruby/we2/leeze.ms'. The active tab shows the command 'thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rspec spec' and its output. The output indicates that the tests passed successfully, with 14 examples and 0 failures, completed in 1.24 seconds. It also shows the randomized seed 16048 and the execution time breakdown: 6.88s user, 0.27s system, 99% CPU, 7.185 total. The prompt 'thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %' is visible at the bottom.

```
thomas@t420: ~/Coding/Ruby/we2/leeze.ms
thomas@t420: rails s
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rspec spec
.....

Finished in 1.24 seconds
14 examples, 0 failures

Randomized with seed 16048

rspec spec 6,88s user 0,27s system 99% cpu 7,185 total
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %
```

Story 5: Refactoring

- Login- und Logout-Test sind nicht DRY.
- Formular ausfüllen und abschicken sollte ausgelagert werden
- Nutzerlogin ist wahrscheinlich für weitere Tests relevant, sollte in mehreren Tests verfügbar sein
- --> Hierfür erstellen wir einen Integration-Test-Helper

Test-Helper

- Methoden, die von mehreren Tests genutzt werden, lagert man in Test Helper aus
- Test-Helper...
 - ... sind einfache Ruby-Module mit Methoden
 - ... befinden sich in spec/support/
 - ... müssen explizit in RSpec eingebunden werden

Story 5: Refactoring

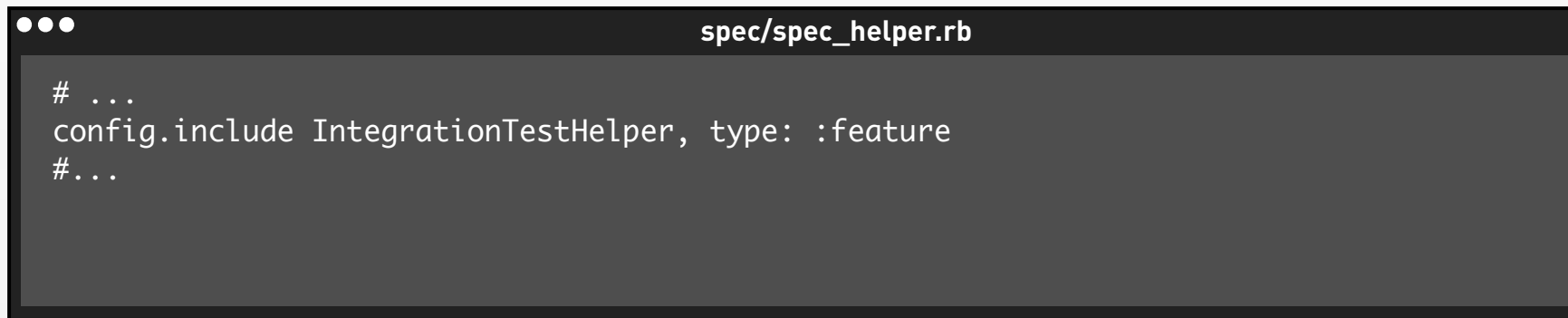
- Test-Helper erstellen

```
spec/support/integration_test_helper.rb

module IntegrationTestHelper
  def sign_in(user)
    click_link 'Sign in'
    fill_in 'user_email', with: user.email
    fill_in 'user_password', with: 'monkey123'
    click_button 'Sign in'
  end
end
```

Story 5: Refactoring

- Test-Helper einbinden

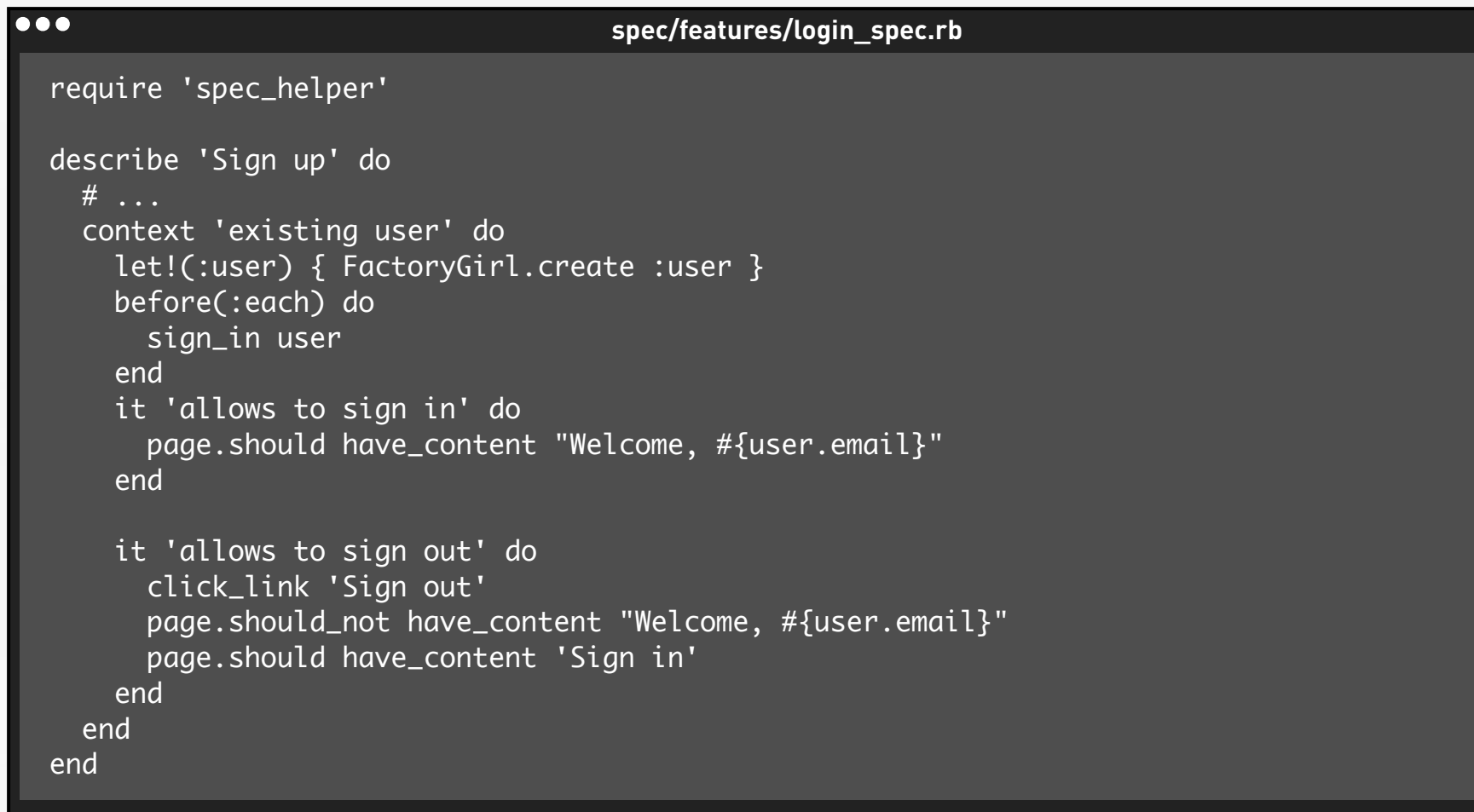
A code editor window with a dark background and a title bar. The title bar contains three small circles on the left and the text 'spec/spec_helper.rb' on the right. The editor area contains the following code:

```
# ...  
config.include IntegrationTestHelper, type: :feature  
#...
```

```
spec/spec_helper.rb  
  
# ...  
config.include IntegrationTestHelper, type: :feature  
#...
```

Story 5: Refactoring

- Test-Helper in Test einbinden



```
spec/features/login_spec.rb

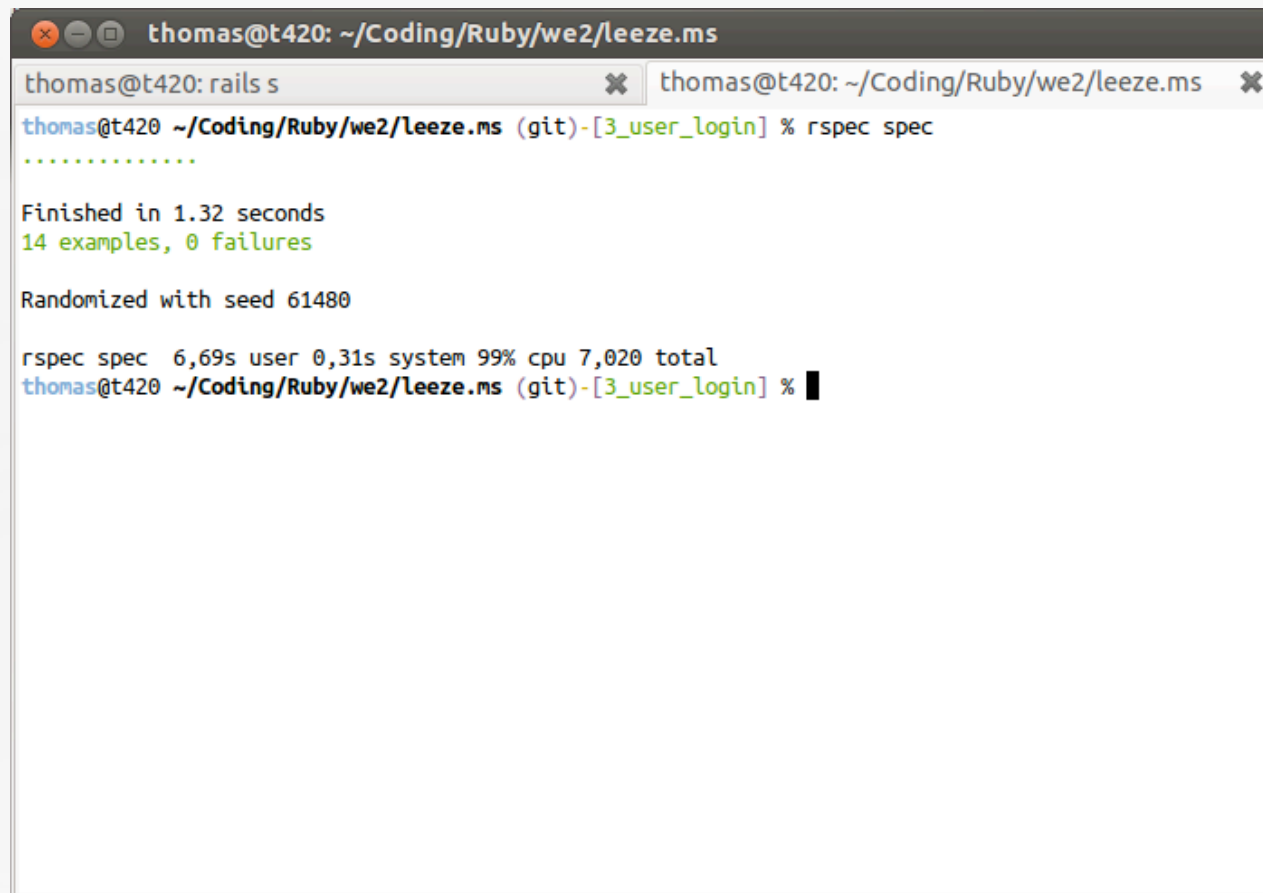
require 'spec_helper'

describe 'Sign up' do
  # ...
  context 'existing user' do
    let!(:user) { FactoryGirl.create :user }
    before(:each) do
      sign_in user
    end
    it 'allows to sign in' do
      page.should have_content "Welcome, #{user.email}"
    end

    it 'allows to sign out' do
      click_link 'Sign out'
      page.should_not have_content "Welcome, #{user.email}"
      page.should have_content 'Sign in'
    end
  end
end
```

Story 5: Refactoring

- Tests ausführen

A terminal window titled 'thomas@t420: ~/Coding/Ruby/we2/leeze.ms' showing the output of running 'rspec spec'. The output indicates that the tests passed successfully. The terminal text is as follows:

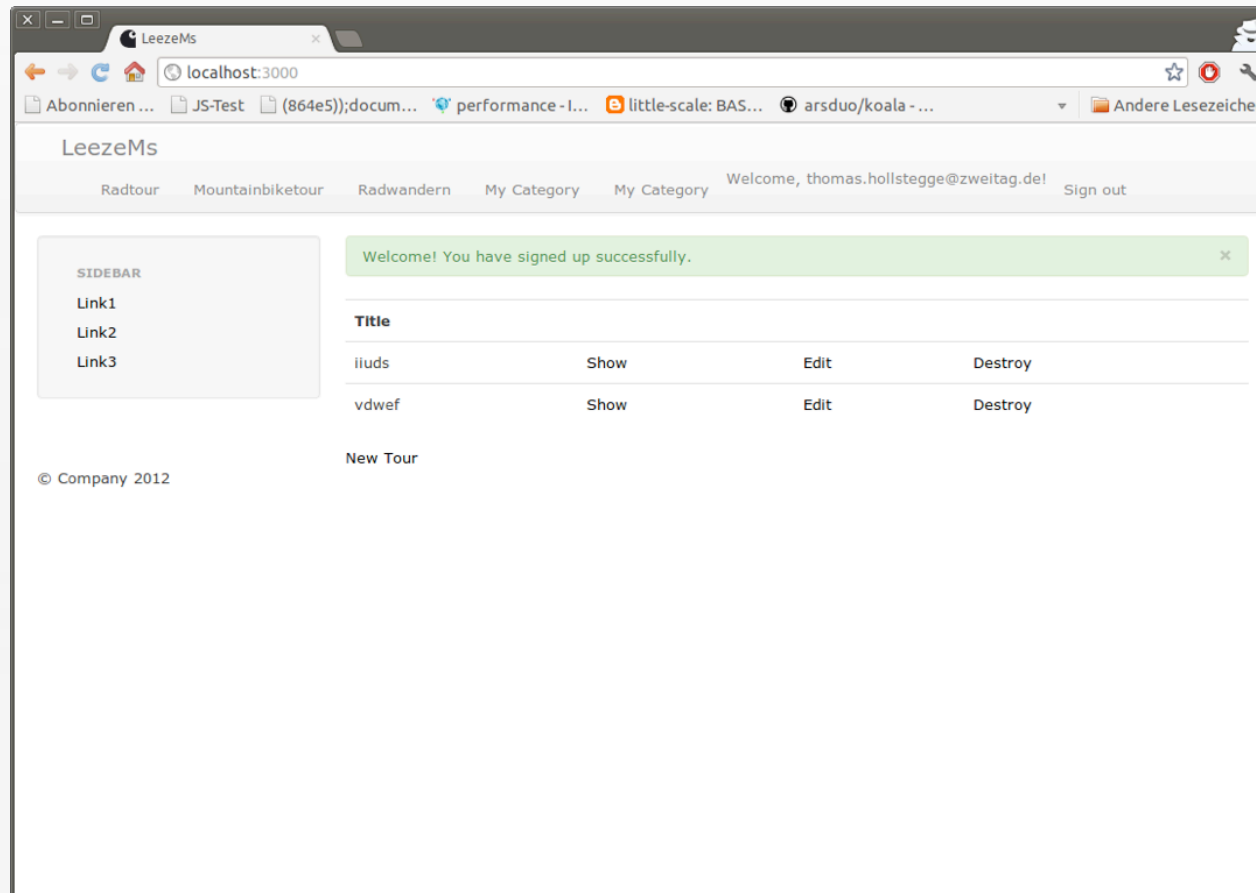
```
thomas@t420: rails s
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rspec spec
.....

Finished in 1.32 seconds
14 examples, 0 failures

Randomized with seed 61480

rspec spec 6,69s user 0,31s system 99% cpu 7,020 total
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %
```


Ergebnis



- --> Navigation fällt auseinander!

Story 5: Refactoring

- Idee: Logout bei Klick auf Mailadresse

```
spec/features/login_spec.rb

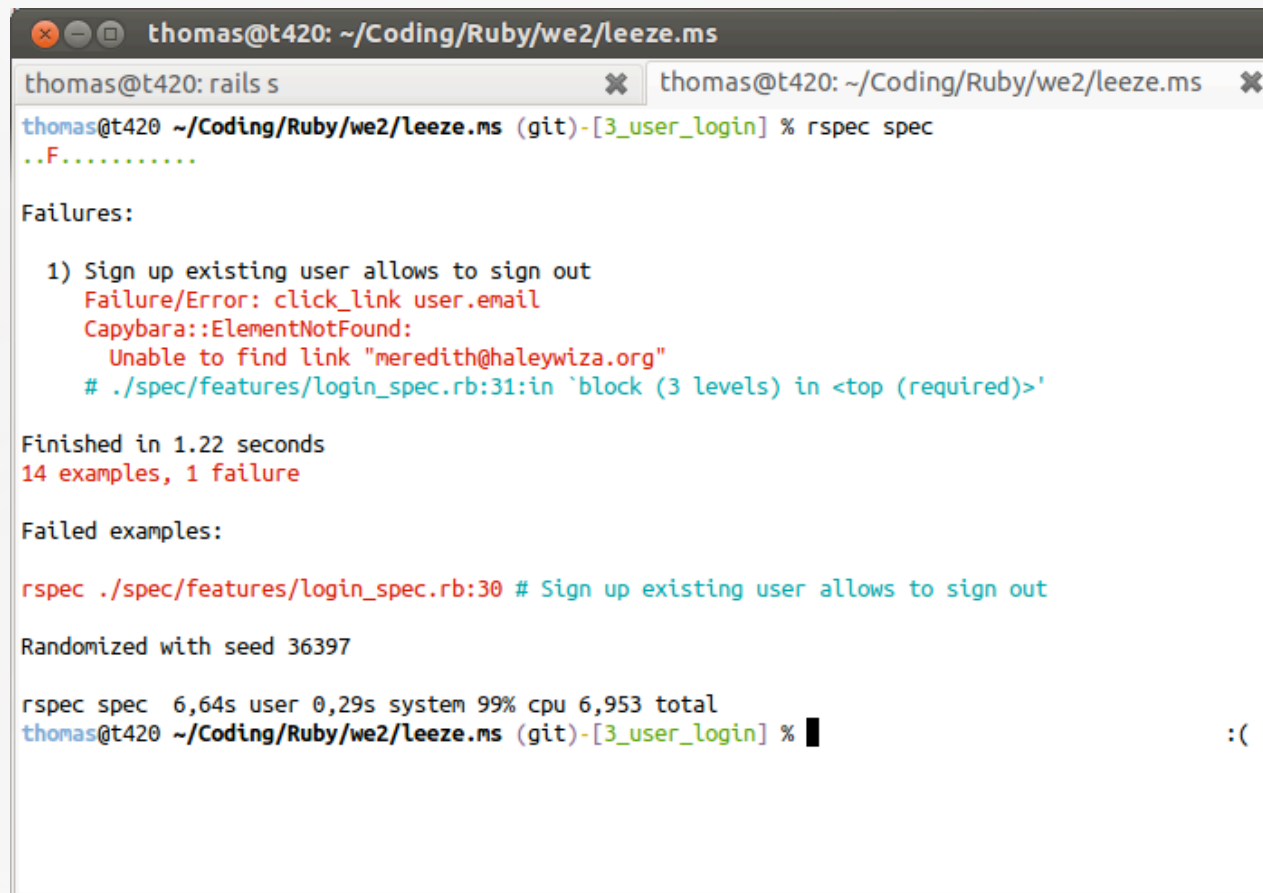
require 'spec_helper'

describe 'Sign up' do
  # ...
  context 'existing user' do
    let!(:user) { FactoryGirl.create :user }
    before(:each) do
      sign_in user
    end
    it 'allows to sign in' do
      page.should have_content "Welcome, #{user.email}"
    end

    it 'allows to sign out' do
      click_link user.email
      page.should_not have_content "Welcome, #{user.email}"
      page.should have_content 'Sign in'
    end
  end
end
```

Story 5: Refactoring

- Tests ausführen



```
thomas@t420: ~/Coding/Ruby/we2/leeze.ms
thomas@t420: rails s
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rspec spec
..F.....

Failures:

  1) Sign up existing user allows to sign out
     Failure/Error: click_link user.email
     Capybara::ElementNotFound:
       Unable to find link "meredith@haleywiza.org"
     # ./spec/features/login_spec.rb:31:in `block (3 levels) in <top (required)>'

Finished in 1.22 seconds
14 examples, 1 failure

Failed examples:

rspec ./spec/features/login_spec.rb:30 # Sign up existing user allows to sign out

Randomized with seed 36397

rspec spec 6,64s user 0,29s system 99% cpu 6,953 total
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % █ :(
```

Story 5: Refactoring

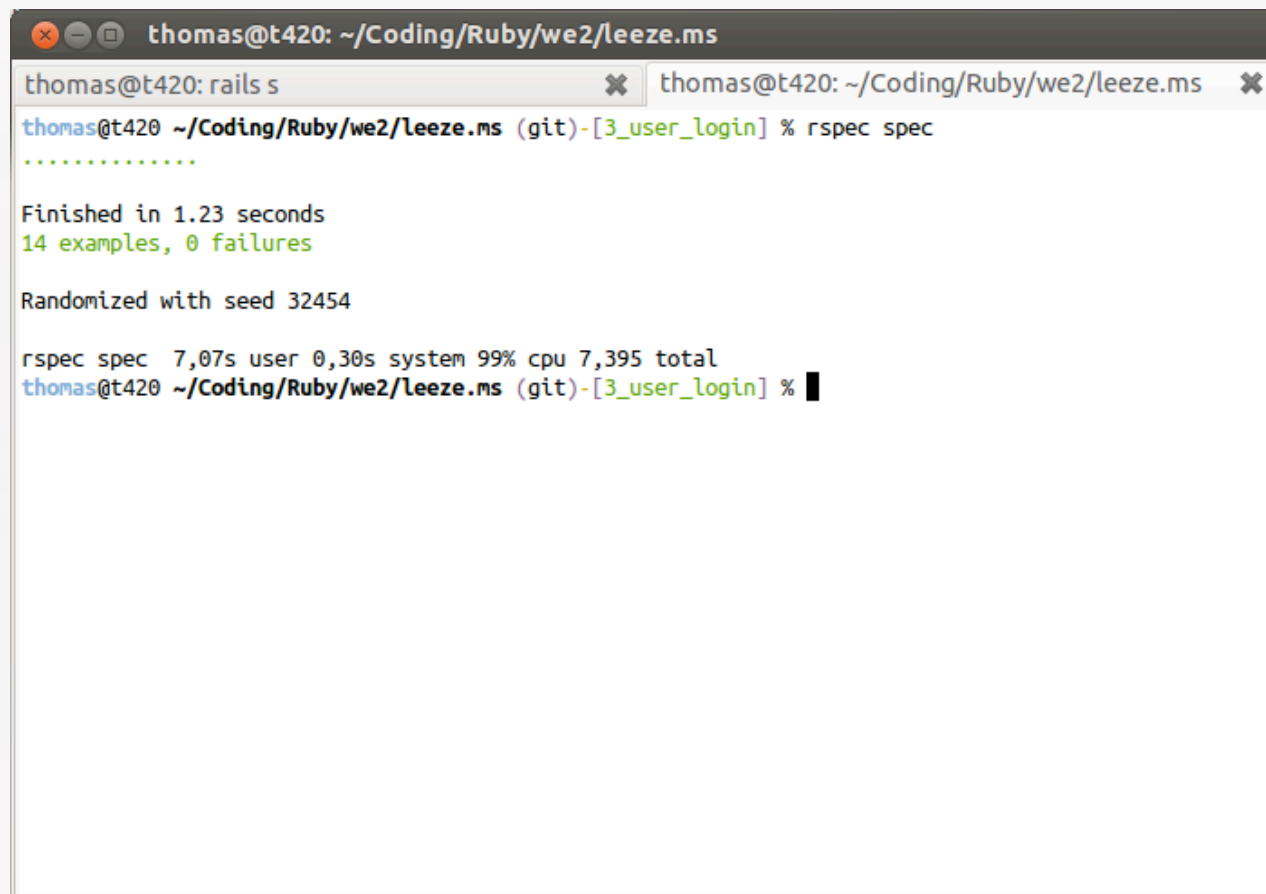
● Navigation anpassen

```
app/views/shared/_navigation.html.erb

<!-- ... -->
<ul class="nav">
  <% Category.all.each do |category| %>
    <li><%= link_to category.name, category_tours_path(category) %></li>
  <% end %>
  <% if current_user %>
    <li>Welcome, <%= link_to current_user.email, destroy_user_session_path,
method: :delete %></li>
  <% else %>
    <li><%= link_to 'Sign in', new_user_session_path %></li>
    <li><%= link_to 'Sign up', new_user_registration_path %></li>
  <% end %>
</ul>
<!-- ... -->
```

Story 5: Refactoring

- Tests ausführen

A terminal window titled 'thomas@t420: ~/Coding/Ruby/we2/leeze.ms' showing the output of an 'rspec spec' command. The output indicates that the tests passed successfully. The terminal text is as follows:

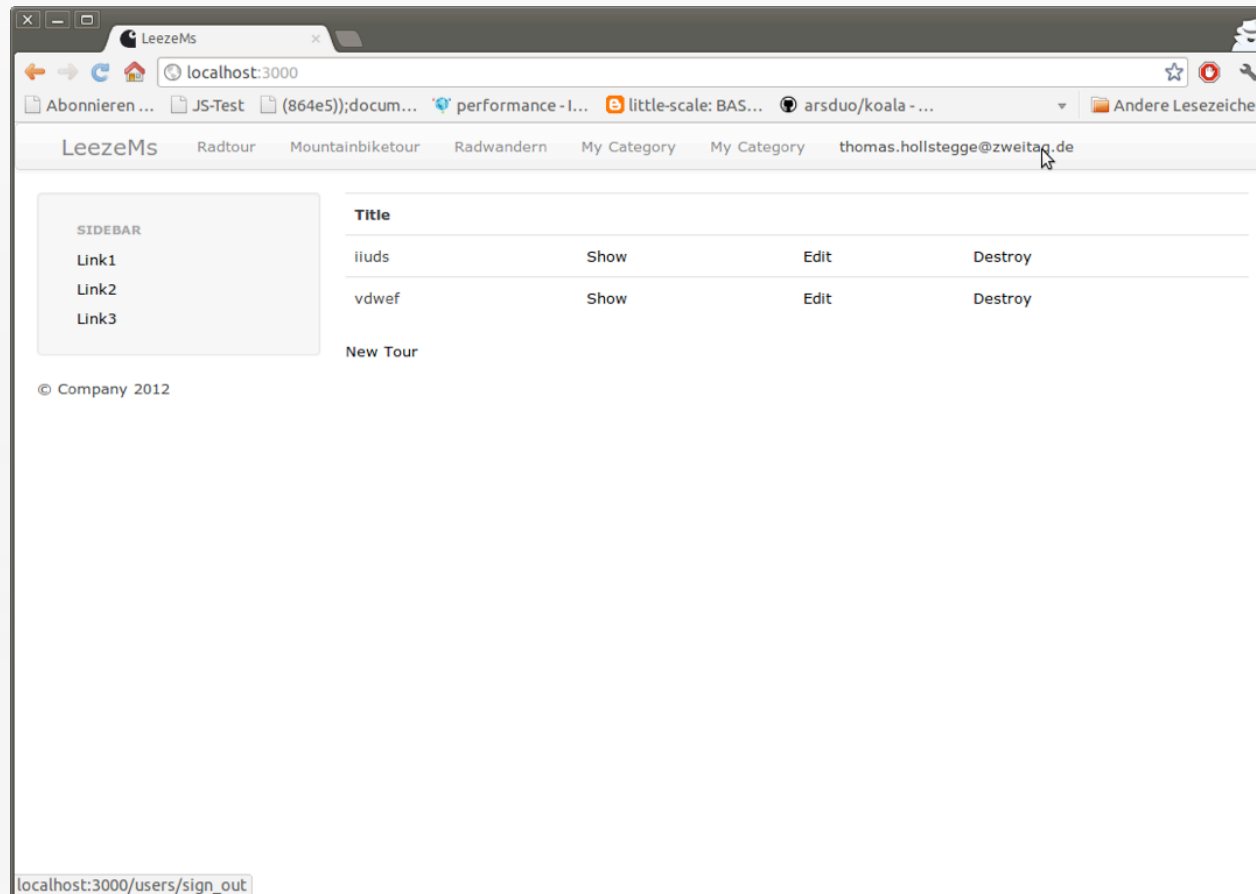
```
thomas@t420: rails s
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rspec spec
.....

Finished in 1.23 seconds
14 examples, 0 failures

Randomized with seed 32454

rspec spec 7,07s user 0,30s system 99% cpu 7,395 total
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %
```

Ergebnis



- --> Weitere Idee: Navigation rechtsbündig machen

Story 5: Refactoring

● Navigation anpassen

```
app/views/shared/_navigation.html.erb

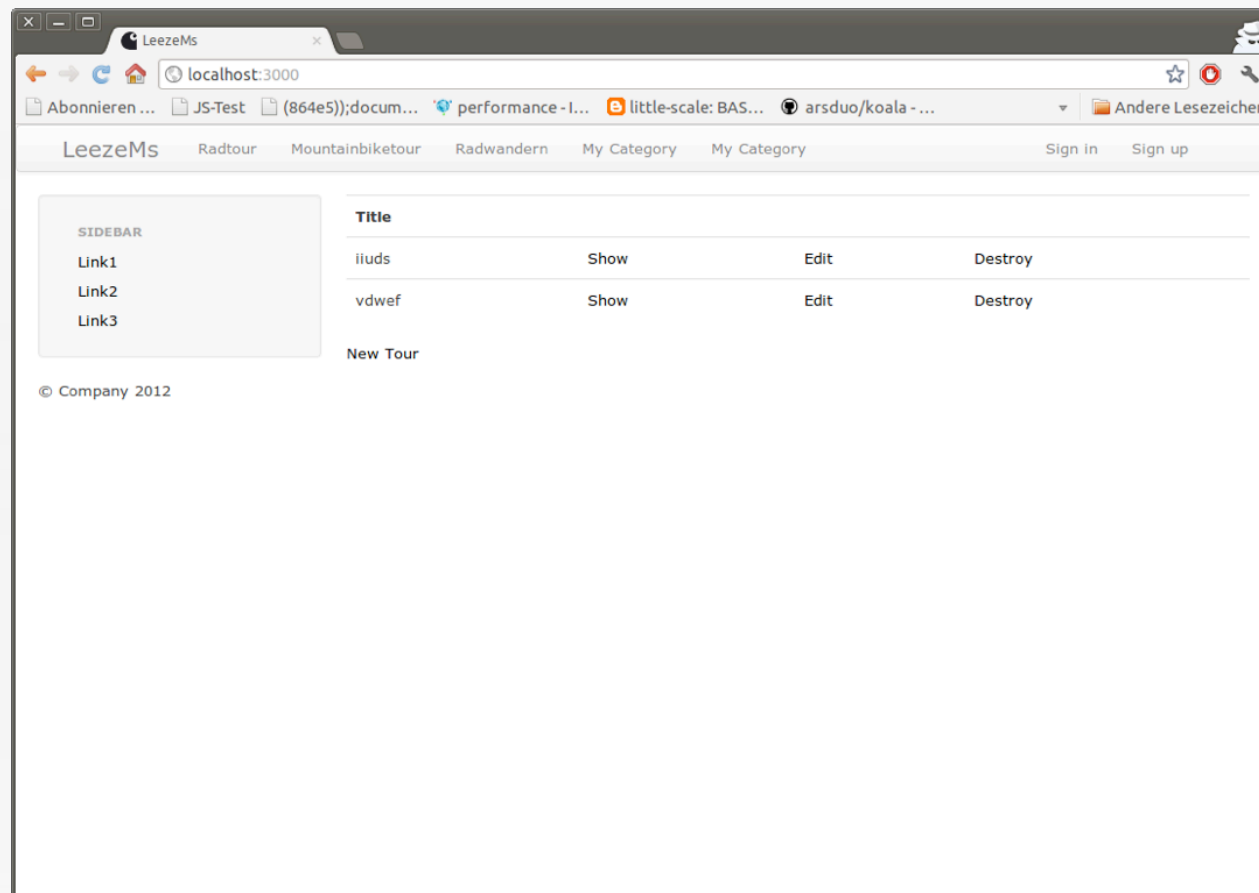
<!-- ... -->
<ul class="nav">
  <% Category.all.each do |category| %>
    <li><%= link_to category.name, category_tours_path(category) %></li>
  <% end %>
</ul>

<ul class="nav pull-right">
  <% if current_user %>
    <li>Welcome, <%= link_to current_user.email, destroy_user_session_path,
method: :delete %></li>
  <% else %>
    <li><%= link_to 'Sign in', new_user_session_path %></li>
    <li><%= link_to 'Sign up', new_user_registration_path %></li>
  <% end %>
</ul>

<!-- ... -->
```

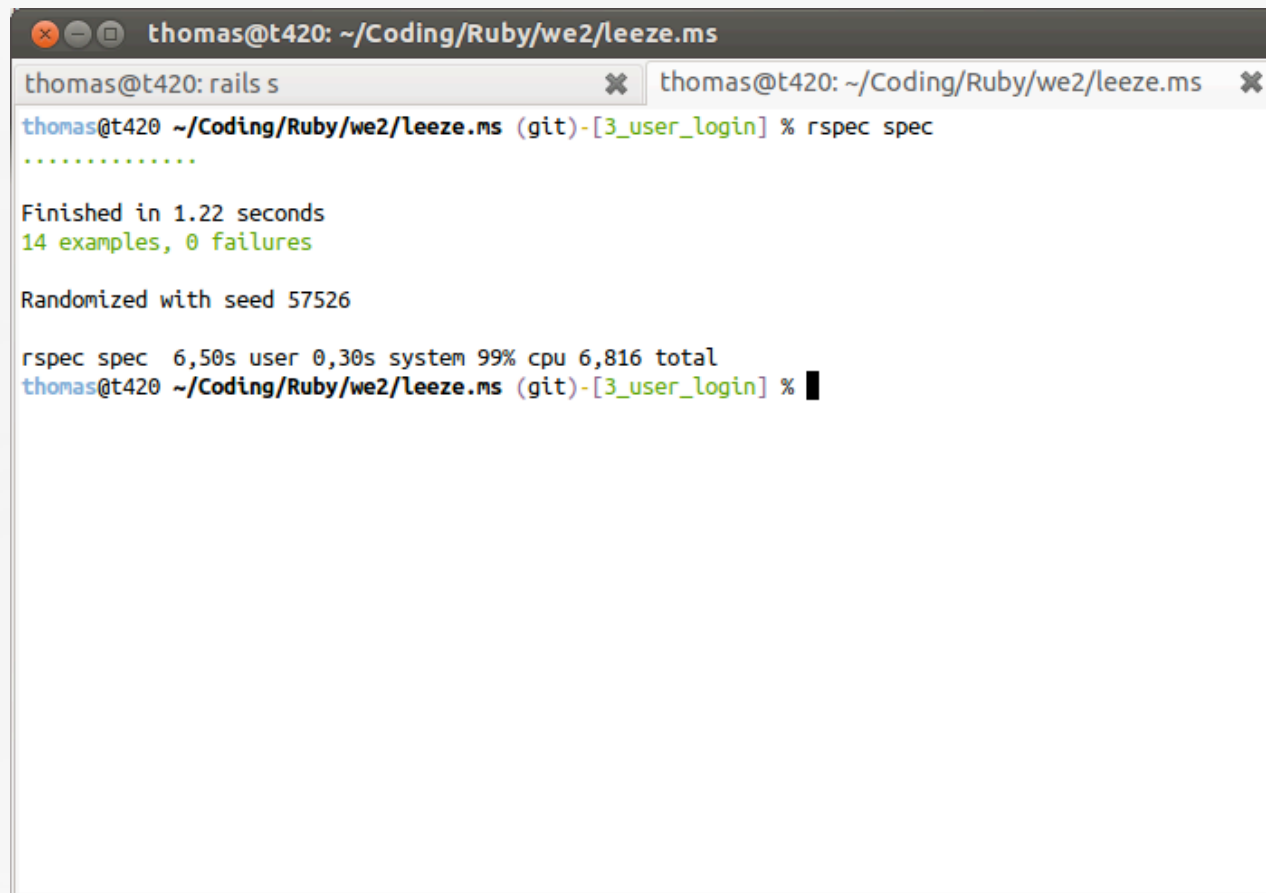
Story 5: Refactoring

- Ergebnis:



Story 5: Refactoring

- Tests ausführen

A terminal window titled 'thomas@t420: ~/Coding/Ruby/we2/leeze.ms' with two tabs: 'thomas@t420: rails s' and 'thomas@t420: ~/Coding/Ruby/we2/leeze.ms'. The active tab shows the command 'thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rspec spec' and its output. The output indicates that the tests passed successfully, with 14 examples and 0 failures, and a randomized seed of 57526. The execution time was 6.50s for the user, 0.30s for the system, and 99% CPU usage. The total time was 6.816s.

```
thomas@t420: ~/Coding/Ruby/we2/leeze.ms
thomas@t420: rails s
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] % rspec spec
.....

Finished in 1.22 seconds
14 examples, 0 failures

Randomized with seed 57526

rspec spec 6,50s user 0,30s system 99% cpu 6,816 total
thomas@t420 ~/Coding/Ruby/we2/leeze.ms (git)-[3_user_login] %
```

Taskboard



Taskboard

