Realizaremos BDD que es el **desarrollo guiado por el comportamiento**.

# TESTING AUTOMATIZADO INTERFAZ WEB feature con 5 escenarios - Gherkin .Feature -> Cucumber -> rb -> framework capybara -> driver selenium -> dsl capybara -> PARA LA RENFE

Documentación: <https://www.solvetic.com/tutoriales/article/2666-testing-web-automatico-con-ruby-capybara-y-selenium/>

Primero instalaremos cucumber, capybara, selenium y rspec:

1. gem install cucumber
2. gem install capybara
3. gem install selenium-webdriver
4. gem install rspec

**GHERKIN:**

Dentro de la carpeta features encontraremos el archivo renfe.feature. Con ello definimos la prueba que queremos pasar en un lenguaje que es entendido por todo el mundo (La prueba en sí es la parte del escenario, el resto es una descripción).

Tenemos 5 escenarios en renfe.feature

**CUCUMBER:**

Dentro de la carpeta features, tenemos dos carpetas step\_definitions y support. (*cucumber –init* también hace esto)

1. Step\_definitions: Aquí encontraremos el vínculo entre cucumber y Gherkin
2. Support: para declarar el entorno (enviroment) y gestionarlo.

Ahora bien, vamos a la cmd de ruby, y dónde tenemos la renfe.feature escribimos “cucumber” que nos muestra *ANEXO1.*

Para rellenar esto, primero tendremos que utilizar capybara y selenium.

Lo rellenamos *ANEXO2.*

Poner cucumber dentro de feature en la cmd de ruby.

**CAPYBARA SINTAXIS PARA LOS STEPS:** <https://gist.github.com/tomas-stefano/6652111>

# TESTING AUTOMATIZADO -SIN-INTERFAZ WEB feature con 5 escenarios (pueden ser los anteriores) -> Gherkin .Feature -> Cucumber -> rb -> framework capybara + phanton -> driver poltergeis -> dsl capybara -> RENFE

Todo lo demás más:

1. gem install poltergeist
2. gem install capybara poltergeist
3. gem install phantomjs

Documentación: <https://github.com/teampoltergeist/poltergeist>

<http://testerstories.com/2014/08/learning-capybara-part-2/>

## ANEXO 0



## ANEXO1

\*\*\* WARNING: You must use ANSICON 1.31 or higher (https://github.com/adoxa/ansicon/) to get coloured output on Windows

Feature: Test Interface

As a client, I the interface to work properly.

Scenario: Change page language # features/renfe.feature:4

Given I am on Renfe in English # features/renfe.feature:5

When you click on Bienvenido # features/renfe.feature:6

Then page language changes to Spanish # features/renfe.feature:7

Scenario: Go to customer service # features/renfe.feature:9

Given the main page # features/renfe.feature:10

When you click Atencion al cliente # features/renfe.feature:11

Then the page goes to customer service # features/renfe.feature:12

Scenario: Change ticket # features/renfe.feature:14

Given the main page # features/renfe.feature:15

When you click Cambios # features/renfe.feature:16

Then the page goes to changes # features/renfe.feature:17

Scenario: Main Page # features/renfe.feature:19

Given the login page # features/renfe.feature:20

When click on Inicio renfe # features/renfe.feature:21

Then access to main page # features/renfe.feature:22

Scenario: Cercanias timeline # features/renfe.feature:24

Given the Cercanias Madrid page # features/renfe.feature:25

When choose origin and destiny, and click Buscar # features/renfe.feature:26

Then trains timeline appears # features/renfe.feature:27

5 scenarios (5 undefined)

15 steps (15 undefined)

0m0.228s

You can implement step definitions for undefined steps with these snippets:

Given("I am on Renfe in English") do

pending # Write code here that turns the phrase above into concrete actions

end

When("you click on Bienvenido") do

pending # Write code here that turns the phrase above into concrete actions

end

Then("page language changes to Spanish") do

pending # Write code here that turns the phrase above into concrete actions

end

Given("the main page") do

pending # Write code here that turns the phrase above into concrete actions

end

When("you click Atencion al cliente") do

pending # Write code here that turns the phrase above into concrete actions

end

Then("the page goes to customer service") do

pending # Write code here that turns the phrase above into concrete actions

end

When("you click Cambios") do

pending # Write code here that turns the phrase above into concrete actions

end

Then("the page goes to changes") do

pending # Write code here that turns the phrase above into concrete actions

end

Given("the login page") do

pending # Write code here that turns the phrase above into concrete actions

end

When("click on Inicio renfe") do

pending # Write code here that turns the phrase above into concrete actions

end

Then("access to main page") do

pending # Write code here that turns the phrase above into concrete actions

end

Given("the Cercanias Madrid page") do

pending # Write code here that turns the phrase above into concrete actions

end

When("choose origin and destiny, and click Buscar") do

pending # Write code here that turns the phrase above into concrete actions

end

Then("trains timeline appears") do

pending # Write code here that turns the phrase above into concrete actions

end

## ANEXO2

Given("I am on Renfe in English") do

visit 'http://www.renfe.com/EN/viajeros/index.html'

end

When("you click on {string}") do |string|

click\_button('Spanish')

end

Then("page language changes to Spanish") do

puts page.inspect

page.should have\_content 'ORIGEN'

end

Given("the main page") do

visit 'http://www.renfe.com/index.html'

end

When("you click {string}") do |string|

click\_button('Atención al cliente')

end

Then("the page goes to customer service") do

puts page.inspect

page.should have\_content 'Departamento'

end

Given("the main page") do

visit 'http://www.renfe.com/index.html'

end

When("you click {string}") do |string|

click\_button('opc4')

end

Then("the page goes to changes") do

puts page.inspect

page.should have\_content 'Buscar'

end

Given("the login page") do

visit 'https://venta.renfe.com/vol/login.do?Idioma=es&Pais=ES&inirenfe=SI'

end

When("click on {string}") do |string|

click\_button('homerenfeventa')

end

Then("access to main page") do

puts page.inspect

page.should have\_content 'ORIGEN'

end

Given("the Cercanias Madrid page") do

visit 'http://www.renfe.com/viajeros/cercanias/madrid/'

end

When("choose origin and destiny, and click {string}") do |string|

fill\_in 'o', with: 'Atocha'

fill\_in 'd', with: 'Mostoles el Soto'

click\_button('Buscar')

end

Then("trains timeline appears") do

puts page.inspect

page.should have\_content 'Solicitados'

end

## ANEXO3

Requite ‘capybara’

require 'capybara/cucumber'

require "selenium-webdriver"

Capybara.default\_driver = :selenium

Capybara.default\_driver = :selenium\_chrome

Capybara.javascript\_driver = :webkit