**Produto**

Login, Perfil do usuário e senha

**Sprint 1**

Estória do Usuario Login

Para que eu possa cadastrar e gerenciar minhas tarefas

Sendo um usuário

Posso acessar o sistema com meu email e senha previamente cadastrado.

Fazer login com email e senha

Clicou em OK, é autenticado com sucesso

Login com sucesso!

Cenario: Usuário deve ser autorizado

Dado que eu acesso a pagina de login

Quando eu faço login com “[felipefprado@hotmail.com](mailto:felipefprado@hotmail.com)” e “123456”

Então devo ser autenticado com sucesso

E devo ver a seguinte mensagem “Login com sucesso”

Cenario: Senha errada

Dado que eu acesso a pagina de login

Quando eu faço login com uma senha errada

Então devo ver a mensagem “Login inválida.”

Cenario: E-mail errada

Dado que eu acesso a pagina de login

Quando eu faço login com uma e-mail errado

Então devo ver a mensagem “Login inválido.”

Cenario: Campo em branco

Dado que eu acesso a pagina de login

Quando clico sem por email ou senha

Então devo ver a mensagem “Campo em branco.”

**Tecnologias Utilizadas**

VSCode, Ruby 2.3 e Cmder 1.3

**Rubygems utilizadas**

gem 'capybara'

gem 'cucumber'

gem 'rspec'

gem 'selenium-webdriver'

Print dos códigos

Funcionalidade: Login

#Sendo um usuário

#Posso acessar o sistema com meu email e senha previamente cadastrado.

#Acesso a pagina de login

Contexto: Pagina de login

Dado que eu acesso a pagina de login

#Usuario cadastrado

@logout

Cenario: Usuário deve ser autorizado

Quando eu faço login com “felipefprado@hotmail.com” e “123456”

Então devo ser autenticado com sucesso

E devo ver a seguinte mensagem “Login com sucesso”

#faz um esquema de cenario para deixar o codigo mais compacto e sem muita repetição

Esquema do Cenario: Tentativa de login

Quando eu faço login com <email> e <senha>

Então devo ver a mensagem <alerta>

Examplos:

    | email                     | senha     | alerta           |

    | felipefprado@hotmail.com  | 12344456  | Login inválida.  |

    |felipefprado@hmail.com     |123456     |Login inválido.   |

    |                           |           |Campo em branco.  |

Código clase “env.rb”

require'capybara'

require 'capybara/cucumber'

#sera aberta a pagina do chrome para os testes

Capybara.configure do |config|

    config.default\_driver = :selenium\_chrome

end

Codigo classe “hooks.rb”

#Fazer logout do usuario

after ('@logout') do

    click\_link 'Sign out'

  end

  # Mostra no html o nome do cenario que foi executado

  after do |scenario|

    nome\_cenario = scenario.name.gsub(' ','\_').downcase

    puts nome\_cenario

  end

Codigos classe “login\_steps.rb

#Ter acesso a pagina login

Dado('que eu acesso a pagina de login') do

    visit 'https://automationpractice.com/index.php?controller=authentication&back=identity'

end

#Pega a classe login da Pages.rb

Quando('eu faço login com {string} e {string}') do |email, senha|

    @email = email

    LoginPage.new.faz\_Login(email, senha)

end

#Ter acesso ao site

Então('devo ser autenticado com sucesso') do

    expect(page).to have\_content @email

end

#Receber a mensagem

Dado('devo ver a seguinte mensagem {string}') do |mensagem|

expect(page).to have\_content mensagem

end

#Fazer logout para testar os outros cenarios para não dar erro

After('@logout') do

   click\_link 'sign out'

end

Codigos classe “pages.rb”

# Faz uma classe com os atributos principais

class LoginPage

    include Capybara::DSL

    def faz\_Login(email, senha)

        find ('imput[name=email]').set email #Pega os imputs da pagina

        find ('imput[name=passwd]').set senha

        click\_button button

    end

end

classe cucumber.yaml

---

default: -p html -p bdd

html: --format html --out=log/features.html #Para executar por padrão em html

bdd: --format pretty # Rodar o relatorio em BDD