

Guia de Automação de Testes com Selenium e xUnit no .NET 8

Este documento descreve como configurar e executar testes automatizados usando Selenium, xUnit e .NET 8 no Visual Studio Code. Vamos criar um teste simples que verifica a conversão de Celsius para Fahrenheit na página de pesquisa do Google.

1. Instalar Visual Studio Code

Baixe e instale o Visual Studio Code do site oficial:

<https://code.visualstudio.com/>

2. Instalar o .NET SDK

Baixe e instale o .NET SDK do site oficial do .NET:

<https://dotnet.microsoft.com/download>

3. Configurar um Novo Projeto .NET

Abra um terminal no Visual Studio Code e crie um novo projeto xUnit:

```
dotnet new xunit -n SeleniumTests  
cd SeleniumTests
```

4. Instalar Pacotes Necessários do NuGet

No terminal, instale os pacotes necessários para Selenium e ChromeDriver:

```
dotnet add package Selenium.WebDriver  
dotnet add package Selenium.WebDriver.ChromeDriver  
dotnet add package xunit.runner.visualstudio  
dotnet add package Microsoft.NET.Test.Sdk  
dotnet add package DotNetSeleniumExtras.WaitHelpers
```

5. Escrever o Código do Teste

Crie um novo arquivo chamado `GoogleCelsiusToFahrenheitTest.cs` no diretório do projeto `SeleniumTests` e adicione o seguinte código:

```

using System;
using Xunit;
using OpenQA.Selenium;
using OpenQA.Selenium.Chrome;
using OpenQA.Selenium.Support.UI;
using SeleniumExtras.WaitHelpers;

namespace SeleniumExamples
{
    public class GoogleCelsiusToFahrenheitTest : IDisposable
    {
        IWebDriver driver;
        WebDriverWait wait;

        public GoogleCelsiusToFahrenheitTest()
        {
            driver = new ChromeDriver();
            driver.Manage().Window.Maximize();
            wait = new WebDriverWait(driver, TimeSpan.FromSeconds(10));
        }

        [Fact]
        public void TestCelsiusToFahrenheitConversion()
        {
            try
            {
                // Navegar para a página de pesquisa do Google
                driver.Navigate().GoToUrl("https://www.google.com/search?q=celsius+to+fahrenheit&oq=celsius+to+&gs_lcrp=EgZjaHJvbWUqCQgAEEUYOxiABDIJCAAQRRg7GIAEMgYIARBFgDkyBwgCEAAyGAQyBwgDEAAyGAQyBwgEEAAyGAQyBwgFEAAyGAQyBwgGEAAyGAQyBggHEEUYPNIBCDIwMzRqMG03qAIIsAIB&sourceid=chrome&ie=UTF-8");

                // Localizar o campo de entrada Celsius usando o atributo apropriado
                IWebElement celsiusInputField =
                wait.Until(ExpectedConditions.ElementIsVisible(By.XPath("//input[@jsname='axsL6b']")));

                // Inserir um valor no campo Celsius
                celsiusInputField.Clear();
                celsiusInputField.SendKeys("100");

                // Localizar o campo de resultado Fahrenheit usando o atributo apropriado
                IWebElement fahrenheitResultField =
                wait.Until(ExpectedConditions.ElementIsVisible(By.XPath("//input[@jsname='fPLMtf']")));

                // Verificar o resultado Fahrenheit
                string result = fahrenheitResultField.GetAttribute("value");
            }
            catch { }
        }

        public void Dispose()
        {
            driver.Quit();
        }
    }
}

```

```

        Assert.Equal("212", result);
    }
    catch (NoSuchElementException e)
    {
        Console.WriteLine("Elemento não encontrado: " + e.Message);
        throw;
    }
    catch (WebDriverException e)
    {
        Console.WriteLine("Erro no WebDriver: " + e.Message);
        throw;
    }
    catch (Exception e)
    {
        Console.WriteLine("Erro inesperado: " + e.Message);
        throw;
    }
}

public void Dispose()
{
    driver.Quit();
}
}
}

```

8. Compilar e Executar os Testes

No terminal, compile e execute os testes:

```

dotnet build
dotnet test

```