



UNIVERSIDAD PRIVADA DE TACNA

FACULTAD DE INGENIERIA

Escuela Profesional de Ingeniería de Sistemas

**INFORME DE LABORATORIO N°04
“ELABORANDO PRUEBAS DE UNA APLICACION
WEB”**

CURSO:

Calidad y Pruebas de Software

DOCENTE:

Ing. Patrick Jose Cuadros Quiroga

ALUMNO:

tu name

(codigo)

Tacna - Perú

2020

Índice

1. OBJETIVOS	2
2. REQUERIMIENTOS	2
3. CONSIDERACIONES INICIALES	2
4. DESARROLLO	3
4.1. Creación del proyecto de prueba de .NET Core	3
4.2. Agregar Selenium al proyecto de prueba	7
4.3. Escribir una prueba de IU usando Selenium	7
4.4. Ejecutar la prueba de IU	7
5. CONCLUSIONES	9
6. RECOMENDACIONES	9
7. WEBGRAFIA	9

ELABORANDO PRUEBAS DE UNA APLICACION WEB

1. OBJETIVOS

– –

2. REQUERIMIENTOS

– Conocimientos

Para el desarrollo de esta práctica se requerirá de los siguientes conocimientos básicos:

• –

– Hardware

• –

– Software

Así mismo se necesitan los siguientes aplicativos

• –

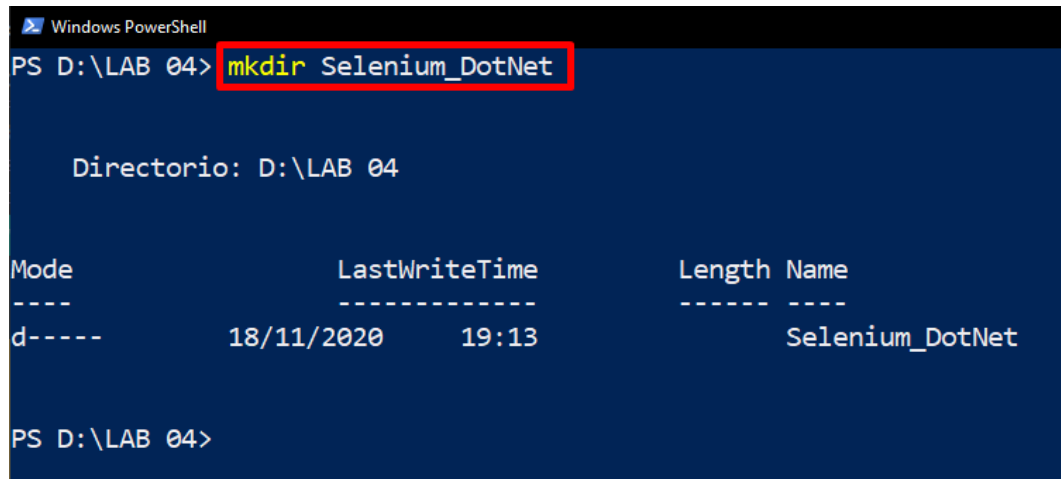
3. CONSIDERACIONES INICIALES

– –

4. DESARROLLO

4.1. Creación del proyecto de prueba de .NET Core

1. En la consola escribir el comando **PS> mkdir Selenium_DotNet** para crear el directorio que albergara la solución.



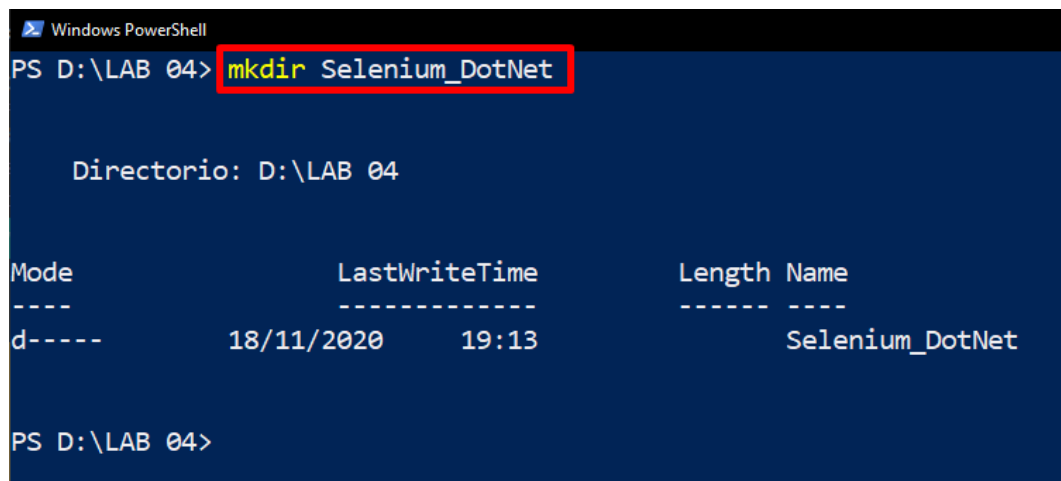
```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

Mode                LastWriteTime         Length Name
----                -
d-----         18/11/2020   19:13                Selenium_DotNet

PS D:\LAB 04>
```

2. Seguidamente ingresar al directorio usando el comando **PS> cd Selenium_DotNet**.



```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

Mode                LastWriteTime         Length Name
----                -
d-----         18/11/2020   19:13                Selenium_DotNet

PS D:\LAB 04>
```

3. Crear la solución usando el comando **PS> dotnet new solution**.

```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

Mode                LastWriteTime         Length Name
----                -
d-----          18/11/2020   19:13             Selenium_DotNet

PS D:\LAB 04>
```

4. Crear el directorio que albergara el proyecto de prueba con el comando **PS> mkdir Demo.SeleniumTests**.

```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

Mode                LastWriteTime         Length Name
----                -
d-----          18/11/2020   19:13             Selenium_DotNet

PS D:\LAB 04>
```

5. Ingresar al directorio creado usando el comando **PS> cd Demo.SeleniumTests**.

```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

Mode                LastWriteTime         Length Name
----                -
d-----          18/11/2020   19:13             Selenium_DotNet

PS D:\LAB 04>
```

6. Crear el proyecto de prueba usando el comando **PS> dotnet new mstest**.

```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

Mode                LastWriteTime         Length Name
----                -
d-----          18/11/2020    19:13             Selenium_DotNet

PS D:\LAB 04>
```

7. Regresar al directorio anterior usando el comando **PS> cd ...**

```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

Mode                LastWriteTime         Length Name
----                -
d-----          18/11/2020    19:13             Selenium_DotNet

PS D:\LAB 04>
```

8. Agregar el proyecto creado a la solución usando el comando **PS> dotnet sln add Demo.SeleniumTests.**

```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

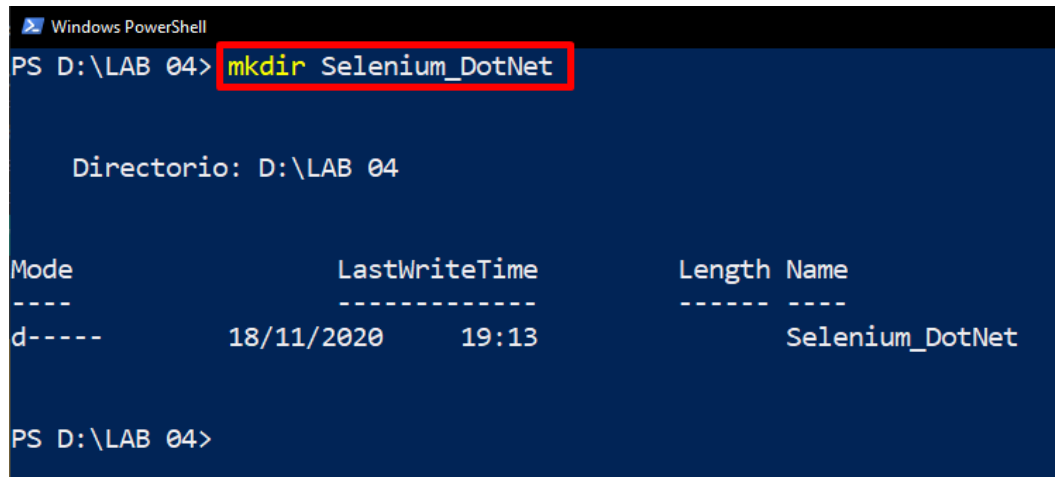
Directorio: D:\LAB 04

Mode                LastWriteTime         Length Name
----                -
d-----          18/11/2020    19:13             Selenium_DotNet

PS D:\LAB 04>
```

Usando los comandos anteriores, el esqueleto del proyecto de prueba está listo.

9. Ingresar al proyecto de prueba con el comando **PS> cd Demo.SeleniumTests**.



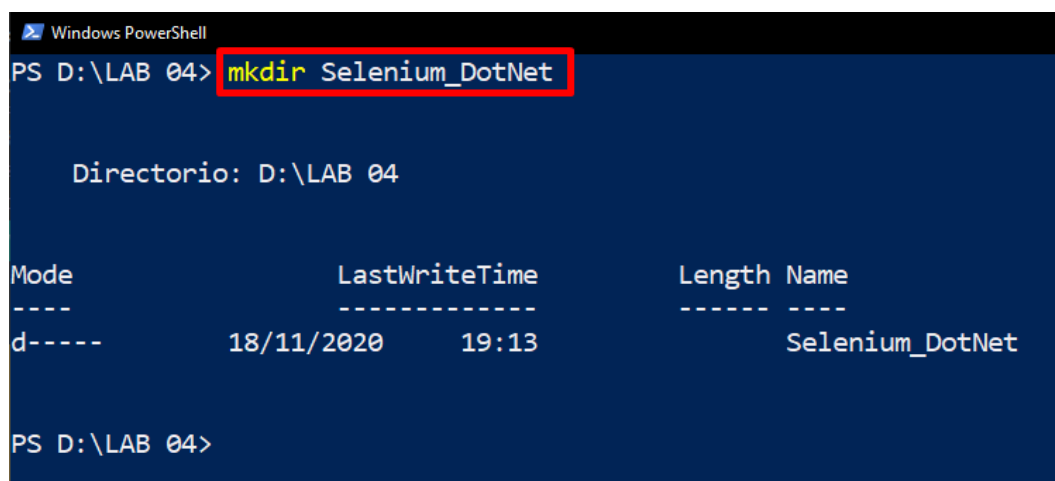
```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

Mode                LastWriteTime         Length Name
----                -
d-----         18/11/2020   19:13                Selenium_DotNet

PS D:\LAB 04>
```

10. Ejecutar el comando **PS> dotnet test** para probar la prueba ficticia.



```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

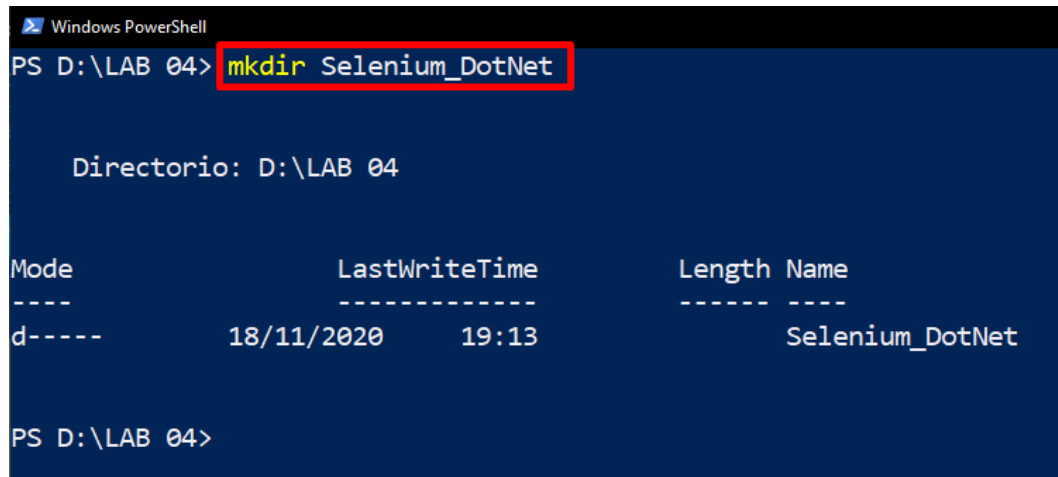
Mode                LastWriteTime         Length Name
----                -
d-----         18/11/2020   19:13                Selenium_DotNet

PS D:\LAB 04>
```

4.2. Agregar Selenium al proyecto de prueba

1. Para comenzar a integrar Selenium en el proyecto de prueba, se debe de agregar el paquete Selenium.WebDriver Nuget.

Usar el siguiente comando *dotnet cli* para agregar la dependencia: **PS> dotnet add package Selenium.WebDriver**.



```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

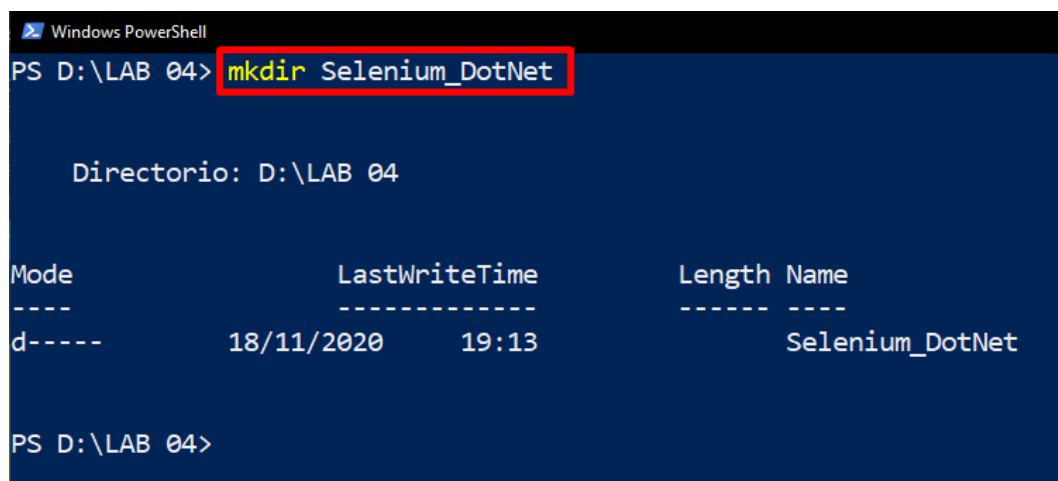
Mode                LastWriteTime         Length Name
----                -
d-----          18/11/2020   19:13             Selenium_DotNet

PS D:\LAB 04>
```

Una vez que se haya agregado el paquete, se podrá usar la API de Selenium en el proyecto C#, pero hay que hacer que **ChromeDriver** esté disponible antes de que comience a funcionar.

4.3. Escribir una prueba de IU usando Selenium

1. Usando el editor de su elección, en este caso **Visual Studio 2019**, escribir una prueba de IU. Mantenerlo simple por ahora, se puede escribir una prueba más avanzada después de hacer que una prueba simple funcione. Para este ejemplo se usara una muestra más avanzada.



```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

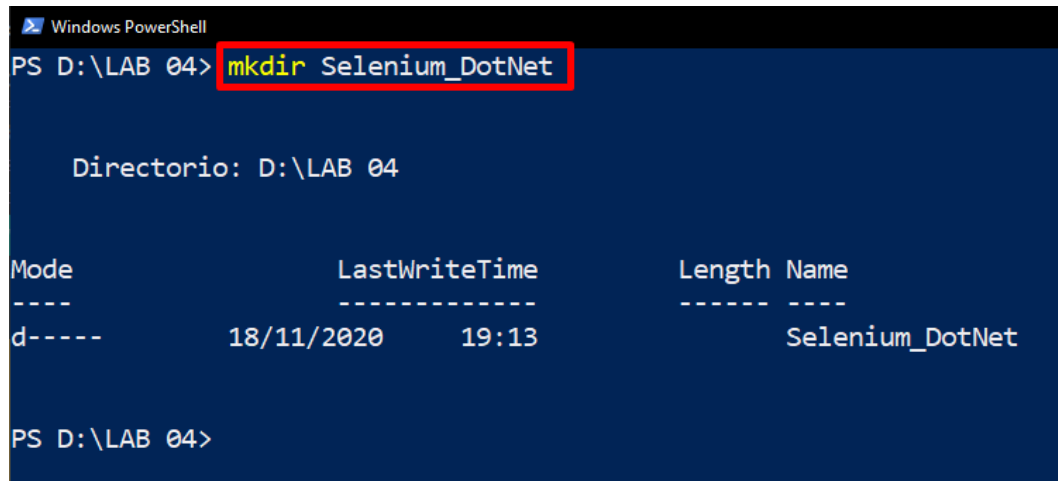
Mode                LastWriteTime         Length Name
----                -
d-----          18/11/2020   19:13             Selenium_DotNet

PS D:\LAB 04>
```

4.4. Ejecutar la prueba de IU

Para ejecutar la prueba, siga estos pasos:

1. Ejecutar el comando **PS> dotnet build** dentro del proyecto de prueba.



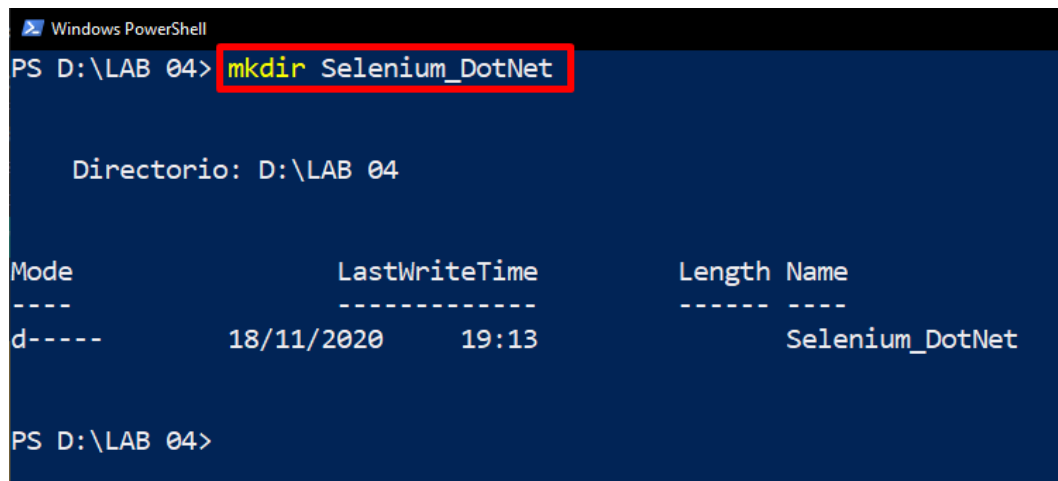
```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

Mode                LastWriteTime         Length Name
----                -
d-----         18/11/2020   19:13                Selenium_DotNet

PS D:\LAB 04>
```

2. Copiar el ejecutable **chromedriver.exe/chromedriver** executable en "*Selenium_DotNet\Demo.SeleniumTests\bin\Debug\net5.0*". En este caso el archivo **chromedriver.exe** se encuentra en descargas así que se tiene que copiar desde **C:\Users\chamo\Downloads\chromedriver.exe** hacia **D:\LAB 04\Selenium_DotNet\Demo.SeleniumTests\bin\Debug\net5.0**. Para realizar la copia se usaría el comando:
PS> cp C:\Users\chamo\Downloads\chromedriver.exe
"D:\LAB 04\Selenium_DotNet\Demo.SeleniumTests\bin\Debug\net5.0".



```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

Mode                LastWriteTime         Length Name
----                -
d-----         18/11/2020   19:13                Selenium_DotNet

PS D:\LAB 04>
```

3. Ejecutar el comando **PS> dotnet test**.

```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

Mode                LastWriteTime         Length Name
----                -
d-----         18/11/2020    19:13             Selenium_DotNet

PS D:\LAB 04>
```

Resultado de la prueba que se realiza.

```
Windows PowerShell
PS D:\LAB 04> mkdir Selenium_DotNet

Directorio: D:\LAB 04

Mode                LastWriteTime         Length Name
----                -
d-----         18/11/2020    19:13             Selenium_DotNet

PS D:\LAB 04>
```

5. CONCLUSIONES

--

6. RECOMENDACIONES

--

7. WEBGRAFIA

--