# Creacion de automatizacion de creacion de Usuarios

Herramientas necesarias instaladas para la realización de la guía:

* Visual Studio Code (u otro editor de código)
* Node.js instalado
* Playwright instalado (npm install playwright)
* Navegador compatible (Chromium, Firefox, WebKit) (npx playwright install)

**Creacion de test:**

Para comenzar la creacion de los test despues de tener el entorno correctamente instalado y actualizado y con nuestro proyecto creado con exito lo primero que hay que hacer es abir la terminal del proyecto en VSCode:

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

En mi caso la carpeta sera D:\Carpeta Nueva Trabajo\Test-elysium

Una vez que estamos situados en la terminal y en la carpeta del proyecto, ejecutamos el siguiente comando:

* npx playwright codegen <URL\_de\_la\_web>

En nuestro caso:

* npx playwright codegen https://elyp.intechsol-pr.net

Este comando abrira Playwright Codegen que es una herramienta que graba las acciones que haces en el navegador y genera el código base para el test.

IMPORTANTE: UNA VEZ EJECUTADO EL COMANDO ANTERIOR LA NUEVA PESTAÑA QUE SE ABRA GRABARA TODAS LAS ACCIONES HECHAS POR EL USUARIO EN DICHA PAGINA POR LO QUE HAY QUE TENER ESPECIAL ATENCION CON LO QUE SE QUIERE GRABAR Y REALIZAR LAS COSAS CORRECTAMENTE PARA EVITAR CODIGO CRUZADO Y ERRONEO.

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

Una vez ejecutado el código se abrirá DOS NUEVAS VENTANAS en nuestro ordenador, por un lado, nos aparecerá la ventana de crohmium la cual será la responsable de captar todo lo que haga el usuario y será transcrito en código en la otra ventana abierta la cual se trata de Playwright Inspector.

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Las primeras líneas de código en aparecer son siempre las importaciones y las funciones correspondientes para realizar las pruebas sin problemas y la primera línea de código de la automatización como tal, la cual se encarga de abrir la url que hayamos puesto.

Comenzamos el test poniendo el usuario y la contraseña, en mi caso:

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

Y comprobamos que se realiza la copia de nuestras acciones en código:

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

Se copia automáticamente nuestra acciones en código, en este caso guarda las credenciales con las que hemos rellenado los apartados de username y password.

Y realizamos lo que queramos automatizar sin problema, en este caso la creacion de un nuevo usuario:

Interfaz de usuario gráfica, Texto, Aplicación, Word

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Realizo la creacion del usuario con total normalidad:

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Codegen recoge todas las acciones en codigo incluso las interacciones con el teclado.

Una vez que finalizamos lo que querremos automatizar le daremos al botón rojo de la parte superior central de la pagina de Crohmium para dejar de grabar las acciones y finalizar el código generado.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

O también podemos hacerlo en la pestaña del inspector para para la grabación:

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Una vez parada la grabación copiamos el código generado y lo copiamos en un nuevo documento que crearemos para la ejecución del test correctamente ***(Correspondiente al proyecto creado)***

En este caso ***creacionusuario.spec.ts***:

Una captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Y pegamos el código generado:

Pantalla de computadora

El contenido generado por IA puede ser incorrecto.

Para no comprometer datos ni del usuario ni de los usuarios creados en el test procederemos a la creacion de de un archivo .env para guardar las credenciales y las urls y ademas crearemos una nueva carpeta llamada data que guardara un archivo .json que contendra los datos del usaurio.

***Creacion .env:***

Lo primero que tenemos que hacer es crear un documento en la raiz del proyecto llamado .env



Dentro del atrchivo lo que tenemos que hacer es definir nuestras variables en este caso: user, password y la url

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

Para poder usarlo tenemos que instalar la librería donentv con el siguiente comando en la consola:

* npm install dotenv

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

Después de la instalación tendremos que poner lo siguiente en el inicio de nuestro test:

* dotenv.config();

Esto es para que el test saque la información del archivo .env

Después de esto creamos una carpeta que se llame ‘data’ y crearemos dentro un archivo .json:

Imagen que contiene Interfaz de usuario gráfica

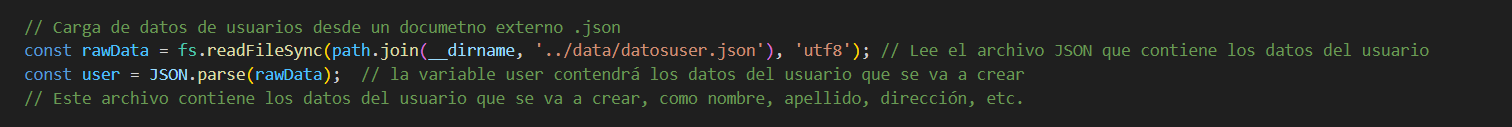
El contenido generado por IA puede ser incorrecto.

Revisaremos los datos rellenados en nuestro documento para organizarlos en el documento .json definidos por una constante:

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

Y para conectar este archivo con nuestro test pondremos lo siguiente:



Esto lo que va a hacer es cargar los datos del documento .json y además lo nombraremos con una variable para dejar nuestro código lo más limpio posible.

Ahora realizamos una revisión del codigo para corregir o eliminar cosas innecesarias como el relleno poco a poco del campo donde añadimos el nombre o doble clicks que haya visto el inspector de código para no tener ningún error para que la realización del test tenga la mejor respuesta posible y que no haya ningún tipo de error.

Ademas ya podemos quitar los datos explícitos del test y poder poner constantes en base de nuestro archivo creado.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Una vez con el código limpio y arreglado ejecutampos nuestro test con el siguiente código:

* npx playwright test \*Nombre\_del\_test\*

En mi caso:

* npx playwright test creacionusuario.spec.ts