

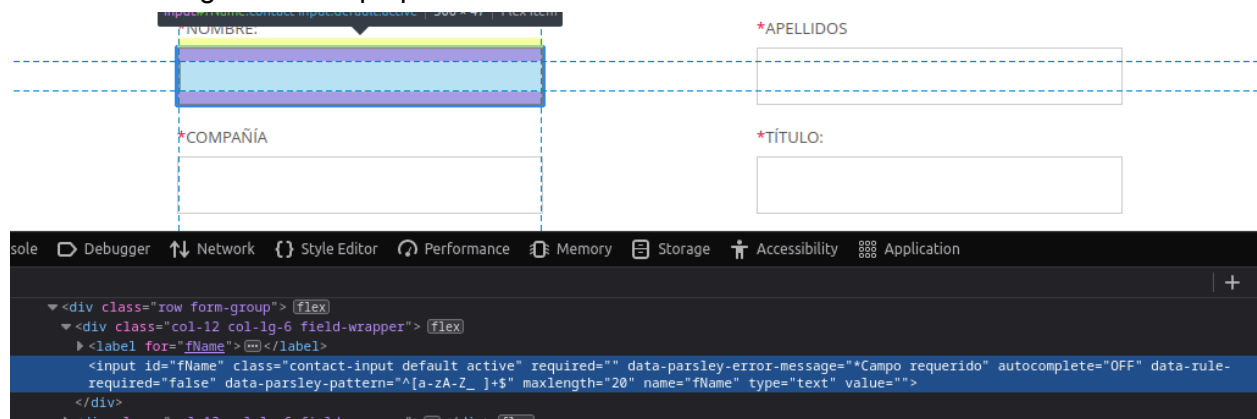
Reporte 2

Breve descripción/demostración del problema.

Dado que este problema puede ser un poco confuso de explicar simplemente con palabras, voy a agregar un par de ejemplos de la vida real en los que este problema se hace presente.

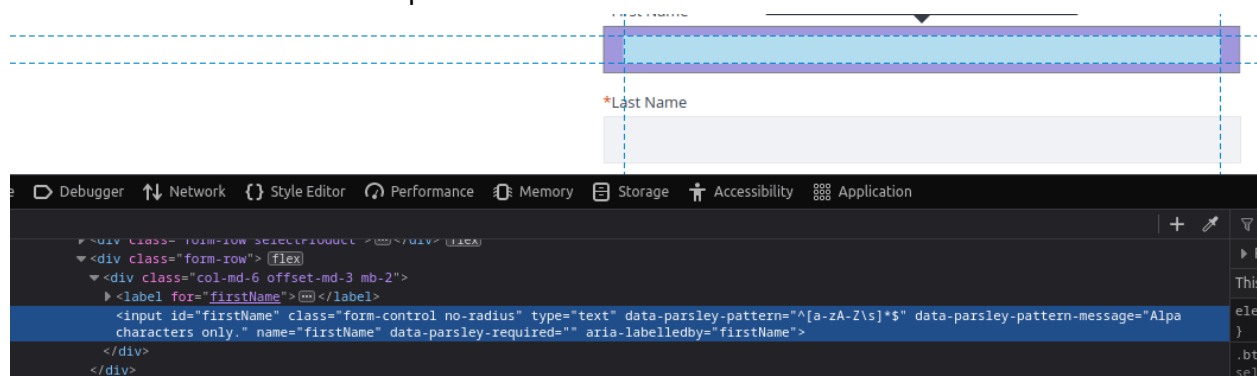
Caso 1: rediseños

Considere el siguiente campo para nombre en un formulario de contacto



Un identificador perfectamente válido para el elemento es su id, que es único y no cambia, salvo que un desarrollador lo modifique.

Ahora considere el mismo campo en el mismo formulario de contacto antes de un rediseño

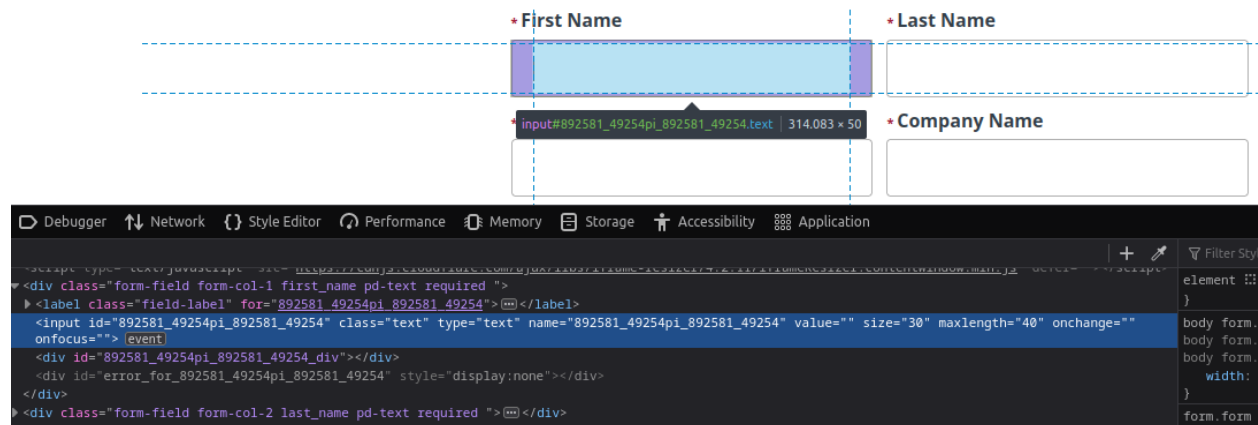


Como se puede observar, el id fue modificado, a pesar de que las funcionalidades y especificaciones del formulario se mantienen. En ocasiones, durante rediseños, se hacen cambios en las convenciones para IDs, clases, etc. A pesar de que estos cambios no afectan la

funcionalidad de los sitios, ocasionan que los scripts de prueba sean incapaces de encontrar los elementos que deben probar.

Caso 2: identificadores por contenido

Considere el siguiente campo para nombre en un formulario de contacto



En este ejemplo es notable que el id del input no es algo que se pueda usar como un identificador con confianza, ya que es generado por una herramienta de generación de contenido. En este caso, la forma que más se acerca a identificar adecuadamente el elemento, es el elemento de tipo input que está después del label que contiene 'First Name'. Sin embargo, esta clase está igualmente sujeta a cambios sin previo aviso, ya que es obtenida a partir del label del input. Como fue mencionado antes, este formulario es generado a partir de contenido dinámico y, por ende, los identificadores son igualmente dinámicos. Si, eventualmente, algún editor cambia el nombre de ese campo de First Name a Name, por ejemplo, el script de pruebas sería incapaz de encontrar el input y la prueba fallaría, a pesar de que ningún cambio fue hecho por el equipo de desarrollo, sino por algún administrador de contenido.

Problema (muy resumido)

A pesar de que es un problema muy simple de resolver y, normalmente, no toma más de unos minutos, tener scripts de prueba que no pasan detiene el pipeline de CI, introduce trabajo no planificado en el sprint (en el caso de metodologías ágiles) y retrasa la integración de funcionalidades nuevas.

Propuesta (muy resumida)

Se quiere investigar si es posible usar técnicas de NLP para identificar un contexto (los elementos alrededor del elemento no encontrado) para predecir qué elemento es el que anteriormente ocupaba ese lugar y sugerir un identificador nuevo.

¿Qué hice esta semana?

Leí los siguientes artículos:

Using Multi-Locators to Increase the Robustness of Web Test Cases

Explica una técnica para intentar solucionar el problema de los identificadores perdidos usando más de un identificador por elemento.

https://www.researchgate.net/publication/273452300_Using_Multi-Locators_to_Increase_the_Robustness_of_Web_Test_Cases

Self Healing Test Automation

Explica el concepto de pruebas que se 'curan' solas usando IA para determinar un nuevo identificador para un elemento.

https://www.researchgate.net/publication/343005103_Self_Healing_Test_Automation

Where do our flaky tests come from?

Una publicación en un blog de Google donde muestran que este es un problema común en sus procesos de desarrollo. Además sugieren que los casos de prueba más grandes son más propensos a tener comportamientos inesperados.

<https://testing.googleblog.com/2017/04/where-do-our-flaky-tests-come-from.html>

Flaky Tests at Google and How We Mitigate Them

Una publicación de Google en la que comentan qué técnicas se usaban en el 2016 para lidiar con tests con comportamiento intermitente y una explicación de porqué la solución utilizada no era óptima.

<https://testing.googleblog.com/2016/05/flaky-tests-at-google-and-how-we.html>

Making System User Interactive Tests Repeatable: When and What Should We Control?

Habla sobre cómo los tests que involucran GUI tienden a tener un alto nivel de entropía y por ende es común que un tests pueda no ser repetible con exactitud. No aporta mucho al proyecto pero se puede considerar como una causa para algunos 'flaky' tests.

<http://www.cs.umd.edu/~atif/pubs/gao-icse15.pdf>

Nótese que muchos de estos artículos son nuevos y no provienen necesariamente de revistas o portales científicos. Probablemente esto se deba a qué es un tema de interés para la industria principalmente, y no para propósitos académicos. Además, fue hasta recientemente que se incorporó el uso de IA en la automatización de pruebas.

¿Qué voy a hacer la próxima semana?

Redactar el problema de manera más elaborada y técnica. Buscar más trabajos relacionados en cuanto a técnicas de identificación y similitud entre elementos HTML.