**APAP QA Coding Challenge**

**Probar los diferentes flujos de trabajo**

https://the-internet.herokuapp.com/login

https://the-internet.herokuapp.com/inputs

https://the-internet.herokuapp.com/hovers

https://the-internet.herokuapp.com/dropdown

**Tareas:**

* **Diseñe un plan de prueba y sus diferentes casos manuales.** 
  + Ver documento Test Plan y Test Suite (Test Cases)
* **Automatice los flujos de trabajo mencionados arriba utilizando Selenium Python, JavaScript o Cypress (Evite utilizar inputs estáticos).** 
  + Ver repositorio de Cypress + JavaScript
    - Index.cy
* **Diseño de componente, regression y smoke test.**
  + **Diseño de componente:**
    - En este caso las smoke test serian parecidas a las unitTest.
  + **Pruebas de regresión:**
    - Esto queda cubierto con la automatización de los flujos de trabajos para una próxima iteración.
  + **Smoke Test:**
    - SmokeTest.cy
* **Crear test suite de las pruebas mencionadas.** 
  + Ver documento de Test Suite
* **¿Cómo probaría en diferentes tamaños de pantalla?** 
  + Usaría el Google Chrome Dev Tools, para simular entornos de web en móviles.
* **¿Cuáles métricas utilizarías?**
  + Pruebas funcionales de los Test Case del Test Suite
    - Resultado de ejecución de los Test Cases
      * Pass vs Fail
    - Defectos reportados
      * Prioridad
      * Estados
      * Defectos desestimados
  + Para el Smoke Test
    - Total de casos ejecutados vs total de casos fallidos
    - Fiabilidad del software, ejecutando varias veces las SmokeTest
      * Cantidad de ejecución sin fallos reportados.
* **Utilice metodología POM (Page Object Model).**
  + Ver Index.cy
* **Suba sus cambios a un controlador de versiones en conjunto con un README explicando como correr las pruebas.**
  + Ver Link de Github

**Compromiso de entrega:**

**17/03/2023 – 28/03/2023**