

CORE DOCUMENTATION

Introducción

Este documento explica y documenta los diferentes folders, librerías y métodos que son utilizados en los scripts de la framework.

Folders creados

* docs: este directorio contiene toda la documentación del proyecto. Arquitectura, las Release Notes, el setup necesario para poder ejecutar los scripts y el ambiente utilizado para ejecutar los tests.
* qa: este directorio contiene todos los documentos pertinentes de QA. El Test Plan, los Test Cases, la Matriz de Trazabilidad y el subdirectorio de reports, que contiene los logs de las test suites ejecutadas.
* src: este directorio contiene todos los scripts, dividido en diferentes folders:
  + lib: contiene diferentes librerías que contienen métodos utilizados en los scripts.
  + scripts: contiene los scripts que tienen los flows que constituyen un test case, o una funcionalidad del teléfono que se está probando. Utilizan los métodos de las librerías.
  + suites: contiene la definición de las test suites, que son esencialmente varios test cases que tienen una funcionalidad en común, y que son llamadas en conjunto para realizar una prueba más robusta.

Métodos creados

Estos métodos son definidos en distintas librerías.

* Logger: esta librería se encarga de mostrar los datos en consola y en archivo:
  + begin\_log(): inicia un nuevo log en el archivo especificado.
  + write\_log(mensaje): escribe en el log y en la consola el menaje recibido.
  + end\_log(): escribe el final de un reporte, con los tiempos inicial y final.
  + error\_log(mensaje): escribe el mensaje de error en el reporte.
* Utils: esta librería abarca pequeños métodos que pueden ser utilizados, pero que no son suficientes para justificar una librería particular.
  + validate\_number(number): realiza la validación de un número de teléfono. Esta validación reconoce entre el número de emergencia 911, números nacionales mexicanos, y números internacionales con el prefijo +.
* PhoneControl: esta librería provee los métodos para conectarse al teléfono a través de adb y UIautomator. También tiene las herramientas para realizar diferentes acciones sobre el dispositivo móvil:
  + read\_serial: identifica el primer número serial de la lista que adb detecte.
  + unlock\_phone: desbloquea el teléfono, mientras no tenga ninguna seguridad activa.
  + click\_home: simula presionar el botón HOME.
  + switch\_button(texto): realiza el click a un switch que contenga el texto pasado por parámetro, y que su className sea android.widget.Switch.
  + click\_button(texto): realiza el click a un botón que contenga el texto pasado por parámetro, y que su className sea android.widget.TextView
  + longclick\_button(texto): realiza el una presión larga a un botón que contenga el texto pasado por parámetro, y que su className sea android.widget.TextView.
  + button\_exists(texto, className): identifica si un botón que contiene el texto y className existe actualmente.
  + click\_detailed\_button(className, packageName, description): realiza el click a un botón que coincida con los tres parámetros recibidos.
  + longclick\_detailed\_button(className, packageName, description): realiza una presión larga a un botón que coincida con los tres parámetros recibidos.
  + detailed\_button\_exists(className, packageName, description): identifica si un botón que contiene los parámetros recibidos existe actualmente.
  + set\_text\_textfield(packageName, contenido): modifica el android.widget.EditText que tenga el packageName recibido como parámetro con el contenido recibido.