

OTO Repair Automation Framework

Installing dependencies:

Python 3.X:

Link to guide: <https://docs.python-guide.org/starting/install3/win/#install3-windows>

Selenium:

Update pip (Python's package management):

- a. open the windows command prompt
- b. type "python -m pip install --upgrade pip"

Install Selenium for Python:

- c. open the windows command prompt (if you closed it before)
- d. type "pip install selenium"

Check Selenium version:

- e. open the windows command prompt (if you closed it before)
- f. type "python -c 'import selenium; print(selenium.__version__)'"

PyTest Framework:

1. Open the Terminal
2. Type "pip install -U pytest"

Framework Structure

The framework follows the POM design pattern and it has the following folders:

1. Pages (where all the methods and locators of specific pages are defined);

2. Automation_tests (where the test cases are defined – and the methods from the pages are are used);
3. Vendor (where the browser drivers are stored).

“Backbone” of the framework is the “**base_page.py**” file.

In that file, all the common methods are defined (this should be continuously upgraded with the desired methods). I’ve added a few ones for finding an element, visiting a page, verifying if an element is displayed, and waiting for an element to be displayed.

In the “contest.py” file I have added the methods for defining the browser (it currently works only for chrome, but it can be easily updated). That file also contains the default URL that you want to use, and the default browser.

In the “config.py” you can set various envs URLs, for example if you want to run the tests on a staging/QA env.

Updates that need to be done

You will have to go to the Automation_test folder and update the following path with your local:

```
25 elif config.browser == "chrome":
26     binary = r'C:\Users\gabrielc\Downloads\OT0_Repair\automation_tests\vendor\chromedriver\chromedriver.exe'
27     driver = webdriver.Chrome(binary)
28     driver.set_window_size(1920, 1080)
29
```

Note: is just that part, and it should point to your project’s root. Don’t remove the “r” or change the /automation_tests/vendor/chromedriver/chromedriver.exe part.

Running the tests

You can simply run the tests once all the dependencies are installed by:

1. Open the terminal into the project’s folder.
2. Type “Py.test”