Front-end Testing

Tim Babych, Sphere Software

https://github.com/tymofij/test_heroes
Installation

• pip install selenium

• Put WebDriver drivers for your browsers into PATH
  • /usr/local/bin
  • GeckoDriver for Firefox
  • ChromeDriver for – duh – Chrome

• Selenium-server is optional
  • PhantomJS RIP
WebDriver spec is W3C candidate recommendation

https://www.w3.org/TR/webdriver/
We now a browser in our IPython shell:

```python
>>> from selenium import webdriver
>>> driver = webdriver.Firefox()

>>> driver.get('http://google.com')
>>> driver.find_element_by_tag_name('title').text
'Google'

>>> q = driver.find_element_by_name('q')
>>> q.send_keys('rimini\n')
```
All their DOM are belong to us now!

We can query elements by
  * CSS
  * XPATH
  * Tag names
  * link text

We can look into value of elements
We can click clicky things
And send keys into typey things
Finders

- `find_element(s)_by_id`
- `find_element(s)_by_name`
- `find_element(s)_by_xpath`
- `find_element(s)_by_link_text`
- `find_element(s)_by_partial_link_text`
- `find_element(s)_by_tag_name`
- `find_element(s)_by_class_name`
- `find_element(s)_by_css_selector`
One finder to rule them all:

from selenium.webdriver.common.by import By

driver.find_element(By.SOMETHING, 'some_rule')
driver.find_elements(By.SOMETHING, 'some_rule')

ID
XPATH
LINK_TEXT
CSS_SELECTOR
XPath selector can be copied from Chrome devtools

And evaluated with $x(query)$
Easy. Like drawing an owl

1. Draw some circles

2. Draw the rest of the sneaky owl
Let’s test an Angular app.

A Tour of Heroes

https://www.npmjs.com/package/angular2-tour-of-heroes
Ugly.py

It is called that way for a reason
Problem with timeouts

• Too little will break things
• Too much will slow tests

• Solution:
  • Wait for condition, not time
  • Implicitly – for all elements
  • Explicitly – only sometimes
This is not the time when dumb tests are ok
Because writing tests must be easy

Otherwise people will not write them
PageObject to the rescue!

Because OOP is Pythonic
It is ok to cut corners

E.g. login is often slow, short-circuit it.
Do not use Selenium

To grab JSON requests is enough
Test one thing at a time

test_search_cart_purchase_delivery_refund()
XPath or CSS?

CSS for everyday, XPath in a clinch
Screenshots
(because looks do matter)

• Needle for Nose
  • https://github.com/python-needle/needle

• Pytest-Needle for Pytest
  • https://github.com/jlane9/pytest-needle
Frameworks
(somebody else’s favorite set of hacks)

• Bok-Choy from EDX
  • https://github.com/edx/bok-choy

• Py.Saunter from Element34
  • https://github.com/Element-34/py.saunter

• Webium from Wargaming
  • https://github.com/wgnet/webium
Thanks!

Questions?
Comments?

Code:
https://github.com/tymofij/test_heroes

http://clear.com.ua

Github: tymofij
Twitter: tymofiy

https://sphereinc.com/