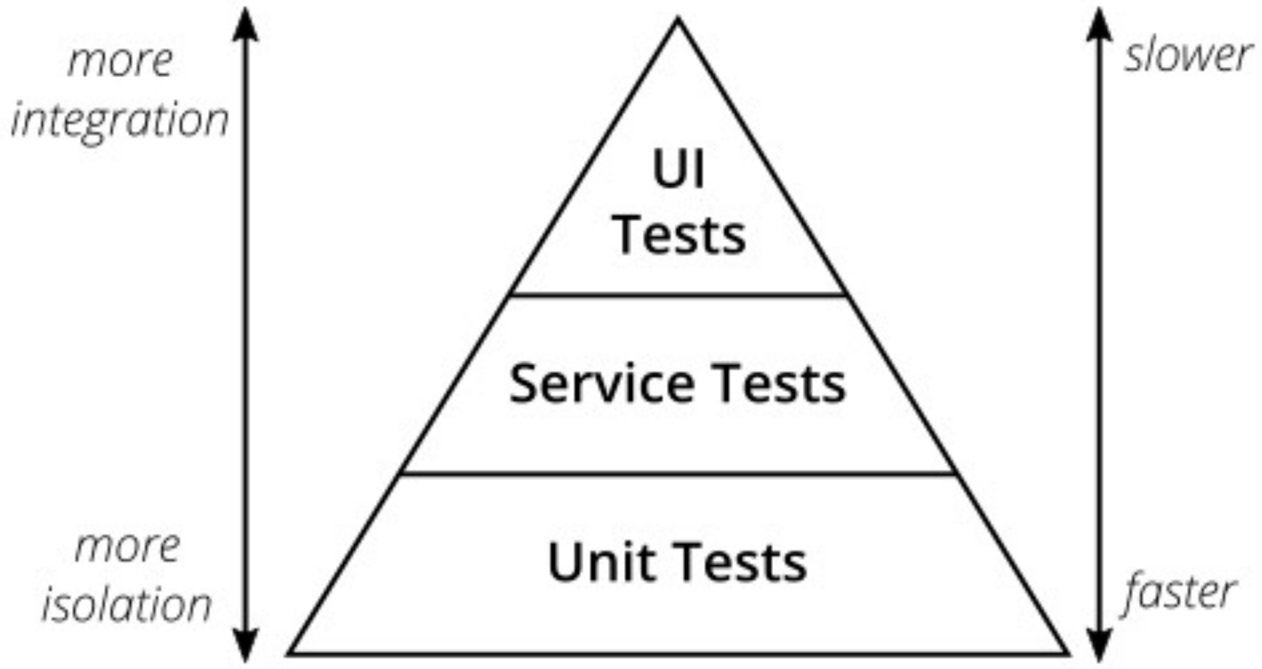
Don’t judge a test by its front end!

Michel Lalmohamed

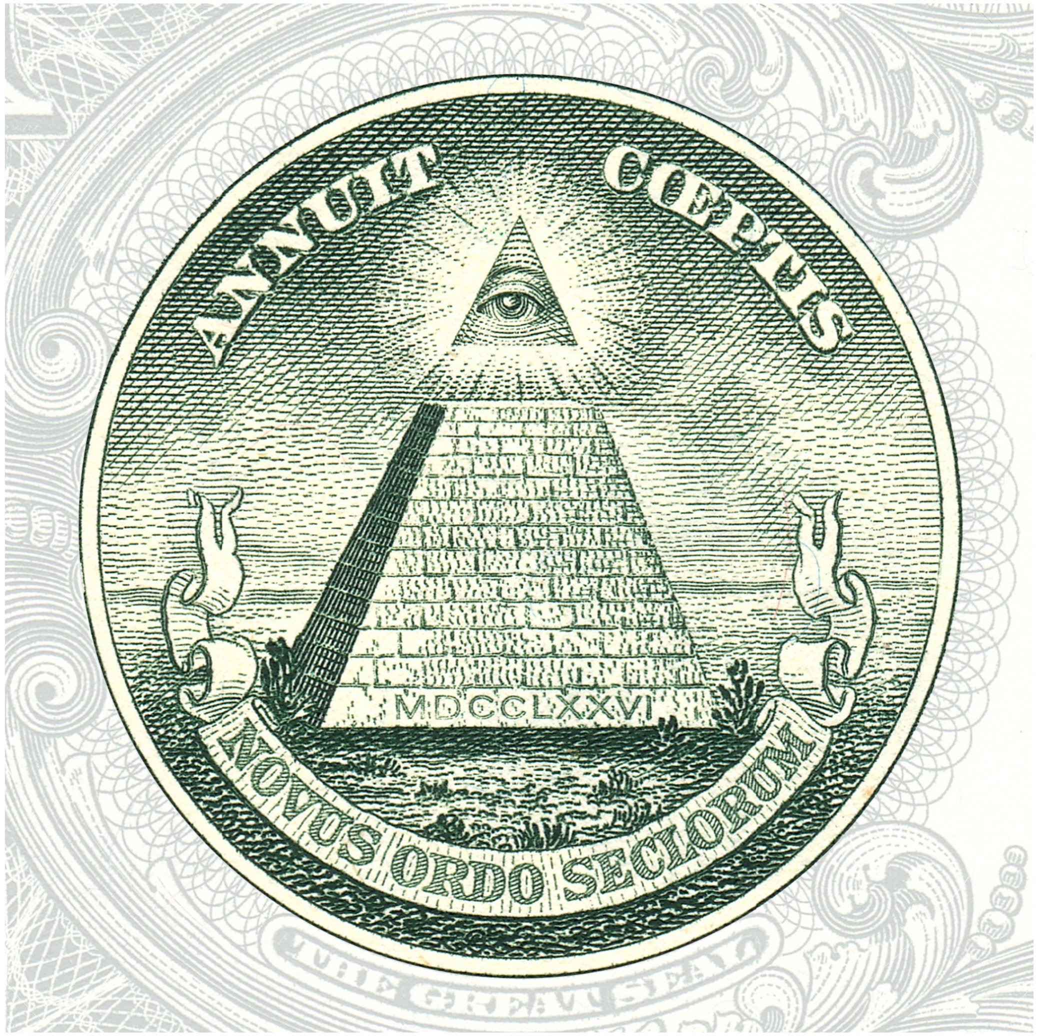
VALORI

Building IT Quality

THE test automation pyramid

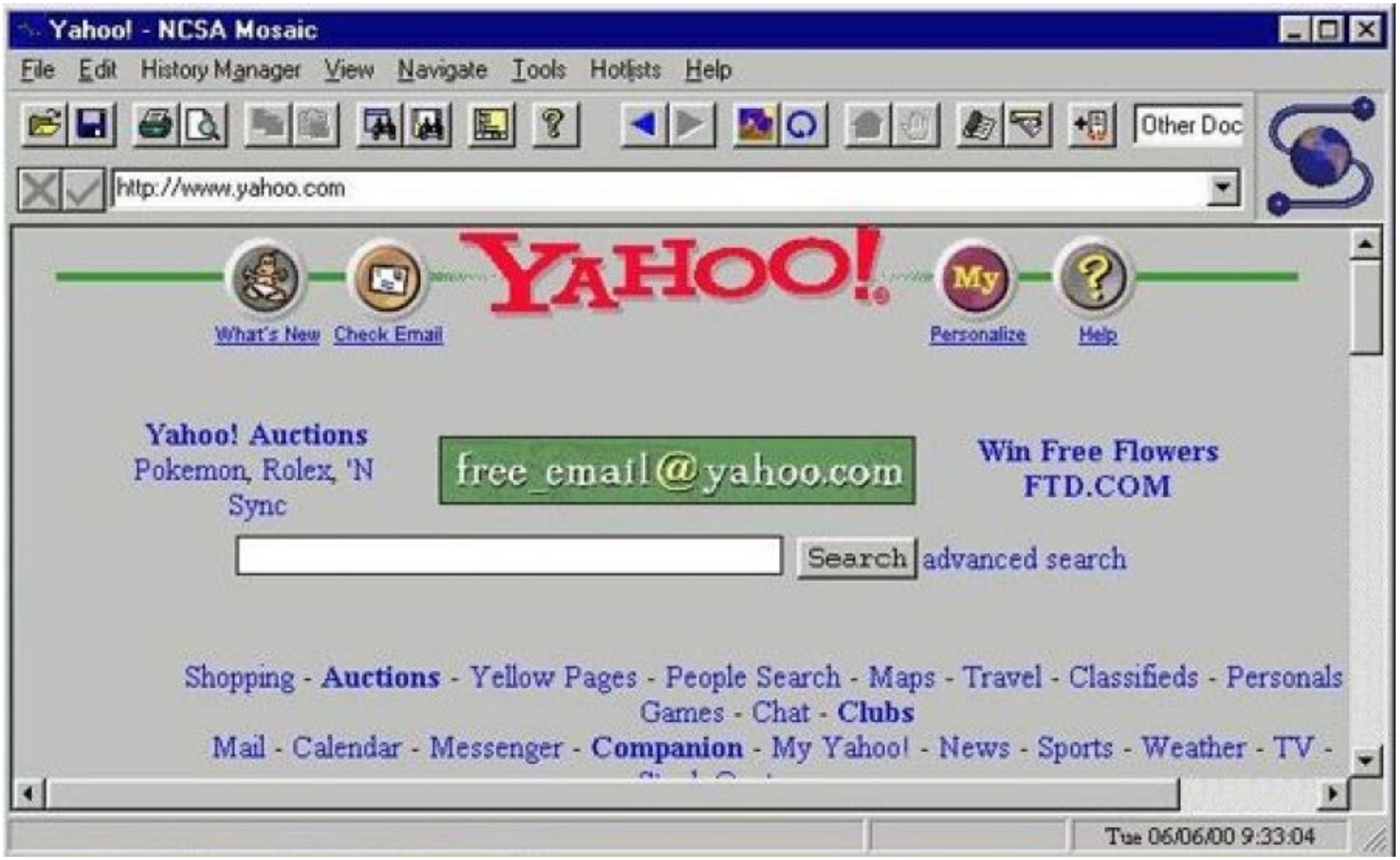


The mighty dollar bill



A relic from the past

UI development & testing


Yahoo! - NCSA Mosaic

File Edit History Manager View Navigate Tools Hotlists Help

Other Doc

http://www.yahoo.com

What's New Check Email YAHOO! Personalize Help

Yahoo! Auctions
Pokemon, Rolex, 'N Sync

free_email@yahoo.com

Win Free Flowers
FTD.COM

Search advanced search

Shopping - **Auctions** - Yellow Pages - People Search - Maps - Travel - Classifieds - Personals
Games - Chat - **Clubs**
Mail - Calendar - Messenger - **Companion** - My Yahoo! - News - Sports - Weather - TV -

Tue 06/06/00 9:33:04

UI development & testing

- Front End Developers
  - **More than backenders**

UI development & testing

- Front End Developers
  - **More than backenders**

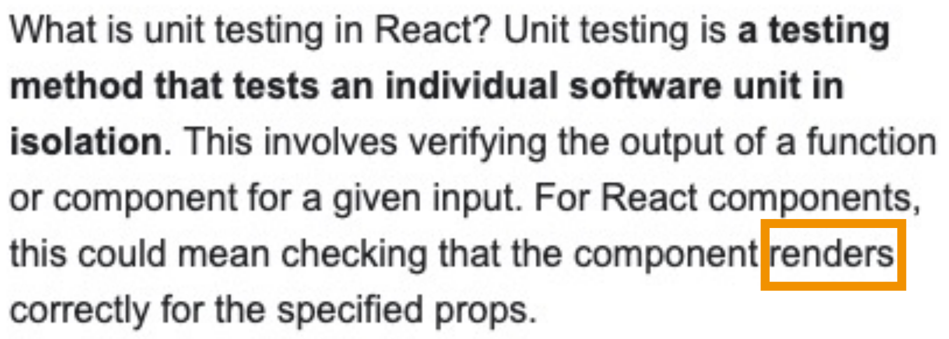
- Front End libraries
  - **React**
  - **Angular**

UI development & testing

- Front End Developers
  - **More than backenders**

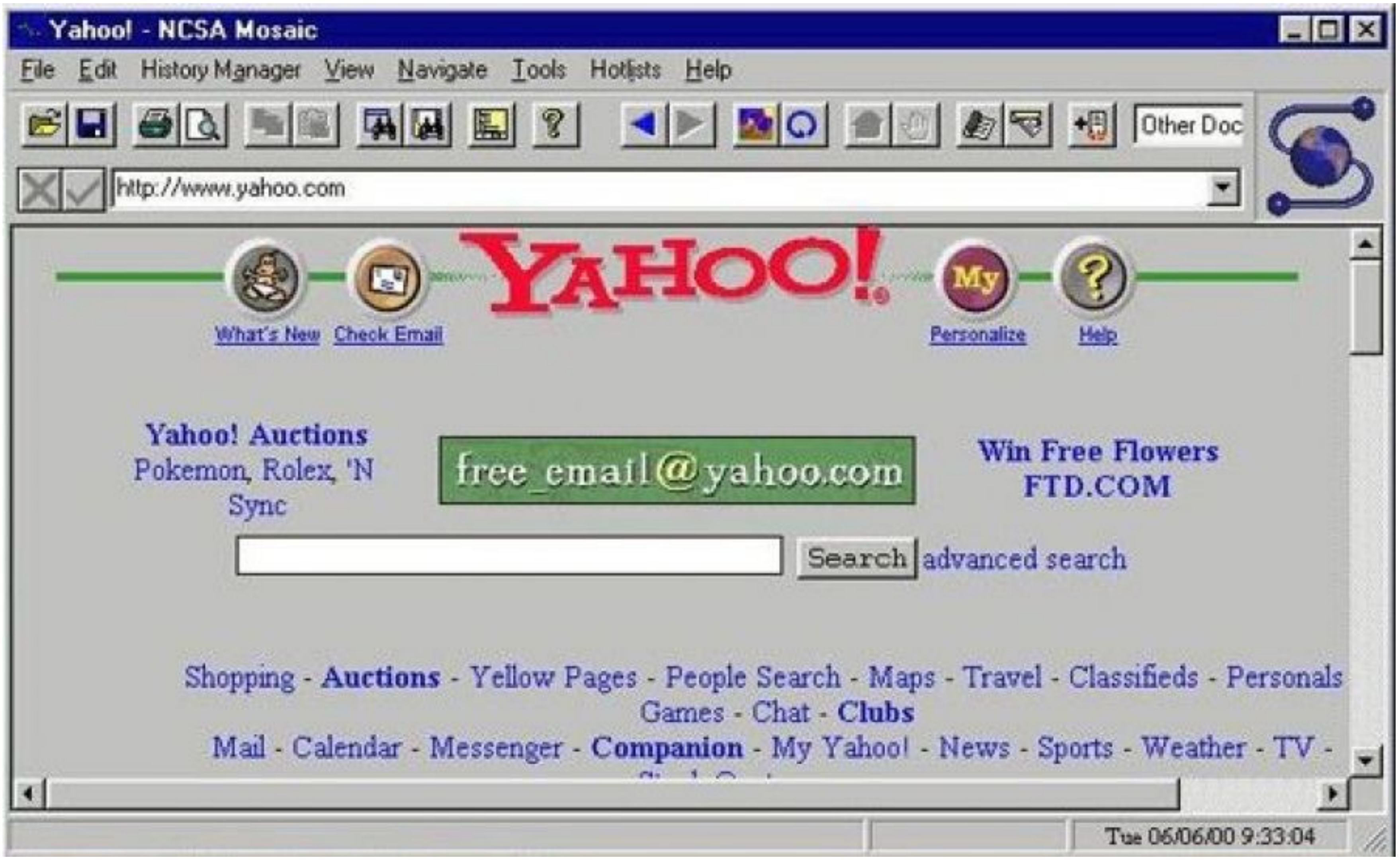
- Front End libraries
  - **React**
  - **Angular**

- Unit test



What is unit testing in React? Unit testing is **a testing method that tests an individual software unit in isolation**. This involves verifying the output of a function or component for a given input. For React components, this could mean checking that the component renders correctly for the specified props.

UI development & testing – started here…



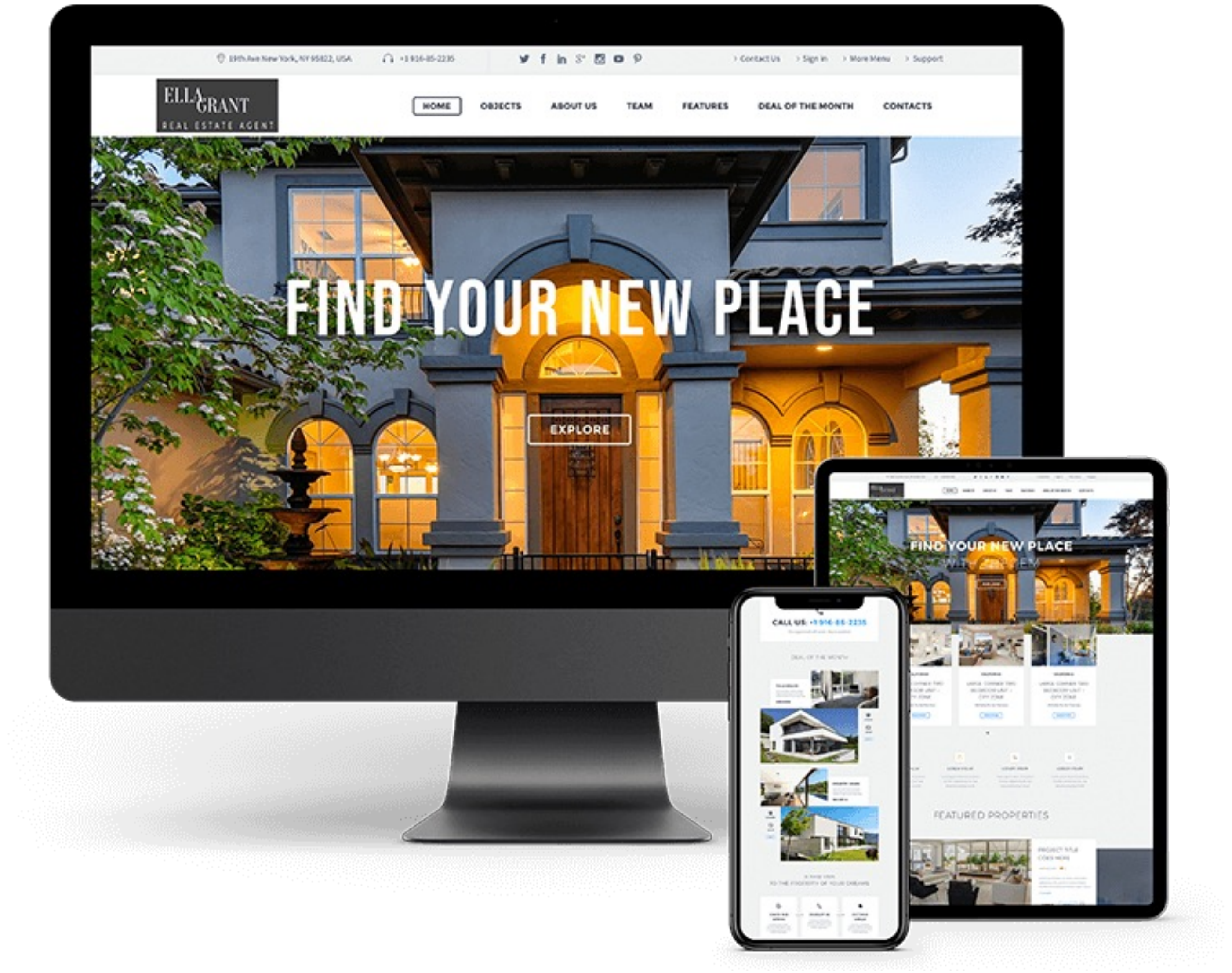
Yahoo! Auctions
Pokemon, Rolex, 'N Sync

free_email@yahoo.com

Win Free Flowers
FTD.COM

Shopping - **Auctions** - Yellow Pages - People Search - Maps - Travel - Classifieds - Personals
Games - Chat - **Clubs**
Mail - Calendar - Messenger - **Companion** - My Yahoo! - News - Sports - Weather - TV -

UI development & testing – now we're here…



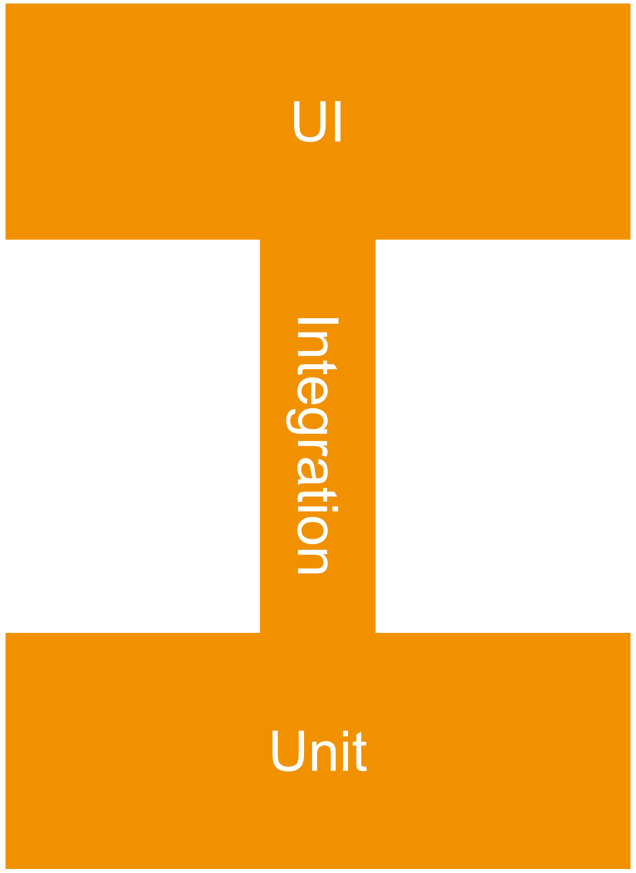
Why keep UI test automation to a minimum by choice?

- Development effort
- Technical complexity

Why keep UI test automation to a minimum as a choice?

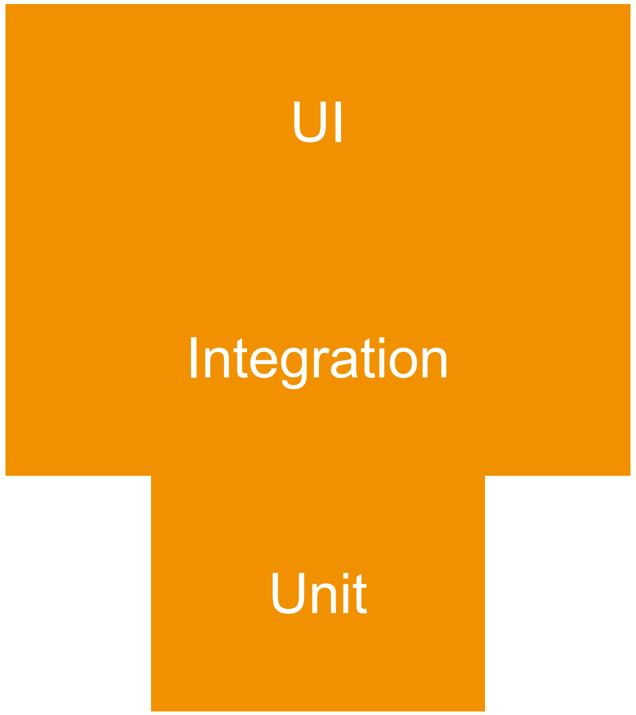
- Development effort
- Technical complexity
- User Interface
  - **U S E R**
- Testing is managing risks, who dares to skip the UI (see point above)?

“I have this model”

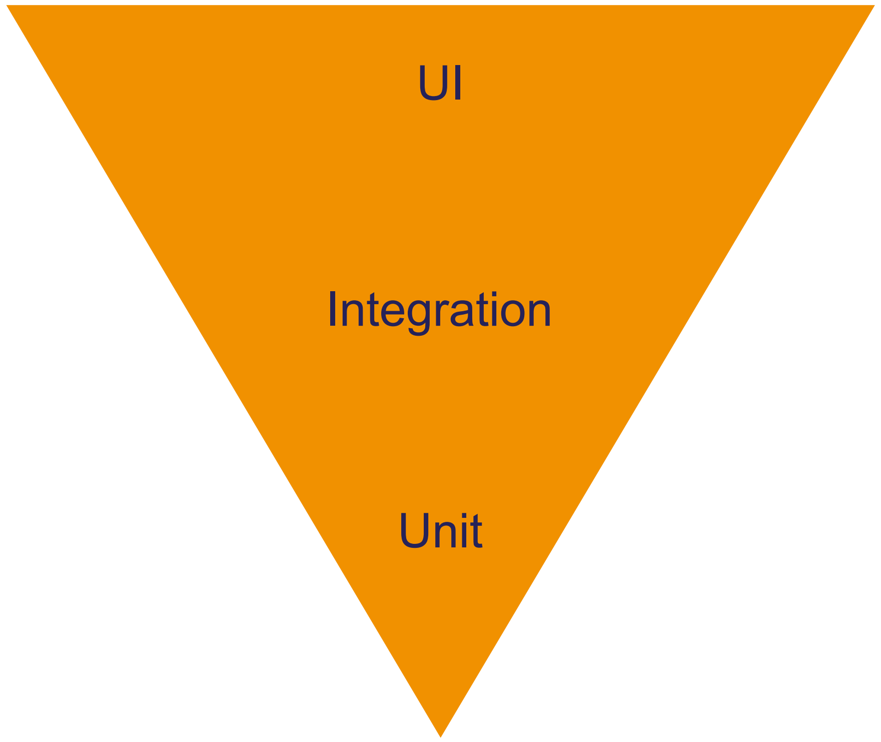


“I have this model”

VALORI



“I have this model”



What if I can’t keep my UI test automation to a minimum?

What if I can't keep my UI test automation to a minimum?

“Your automation effort will be expensive, slow and flaky!” *

* = This message is not supported, approved or paid for by Michel Lalmohamed

EXPENSIVE

SLOW

FLAKY

Bye!

Expensive, slow and flaky UI tests

- UI tester is more expensive?
- Selenium WebDriver is more expensive than Postman?
- GraphQL cheaper then React?

Expensive, slow and flaky UI tests

- UI tester is more expensive than a backend tester?
- Selenium WebDriver is more expensive than Postman?
- GraphQL cheaper than React?

- Development cycle
  - **Finding an issue sooner is cheaper**
  - **Backend development is ahead of front-end development!**

Expensive, slow and flaky UI tests

- UI tester is more expensive than a backend tester?
- Selenium WebDriver is more expensive than Postman?
- GraphQL cheaper than React?

- Development cycle
  - **Finding an issue sooner is cheaper**
  - **Backend development is ahead of front-end development!**

- Test the code with unit testing!

- Test the integration layer with integration testing!

Expensive, <u>slow</u> and flaky UI tests

- Compared to?
  - **Unit**
  - **Integration**

Expensive, slow and flaky UI tests

- Compared to?
  - **Unit**
  - **Integration**

- Path of the end user
  - **Quicker than manual**
  - **Reliable**

Expensive, slow and flaky UI tests

- Compared to?
  - **Unit**
  - **Integration**

- Path of the end user
  - **Quicker than manual**
  - **Reliable**

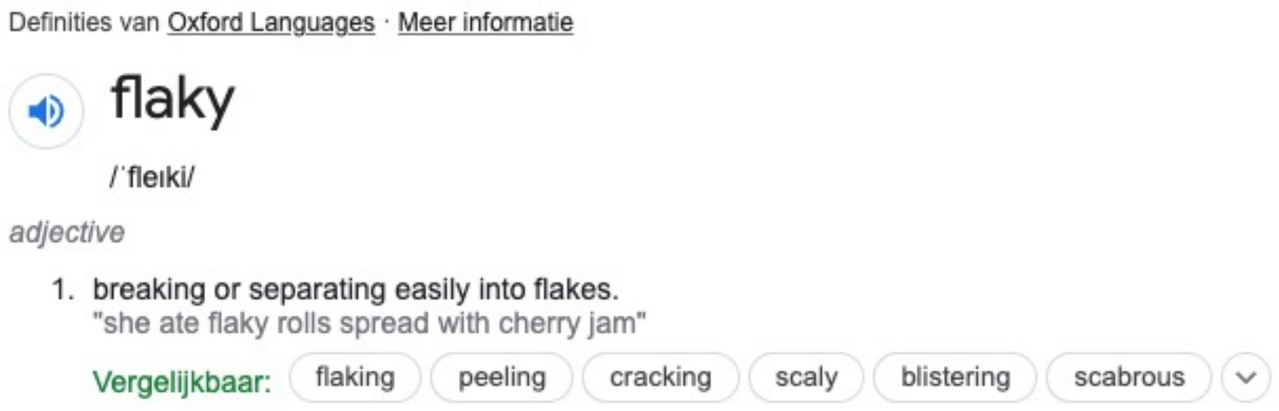
- In the eye of the beholder
  - **It’s all relative!**

Expensive, slow and <u>flaky</u> UI tests

- What is flaky?

Expensive, slow and <u>flaky</u> UI tests

- What is flaky?
  - **Breaking**
  - **Inconsistent**



Definities van Oxford Languages · Meer informatie

flaky

/ˈfleɪki/

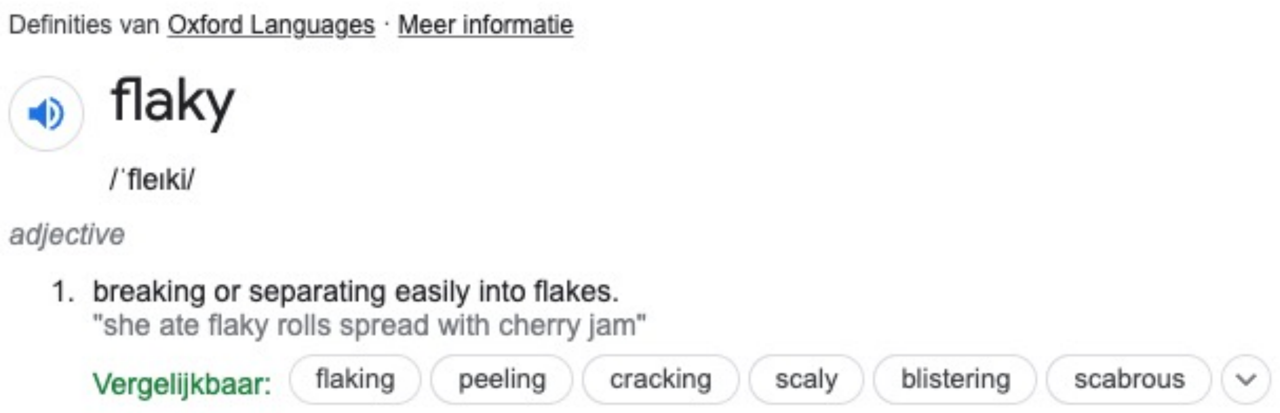
adjective

1. breaking or separating easily into flakes.
   "she ate flaky rolls spread with cherry jam"

   Vergelijkbaar: flaking peeling cracking scaly blistering scabrous

Expensive, slow and <u>flaky</u> UI tests

- What is flaky?
  - **Breaking**
  - **Inconsistent**



Definities van Oxford Languages · Meer informatie

flaky

/ˈfleɪki/

adjective

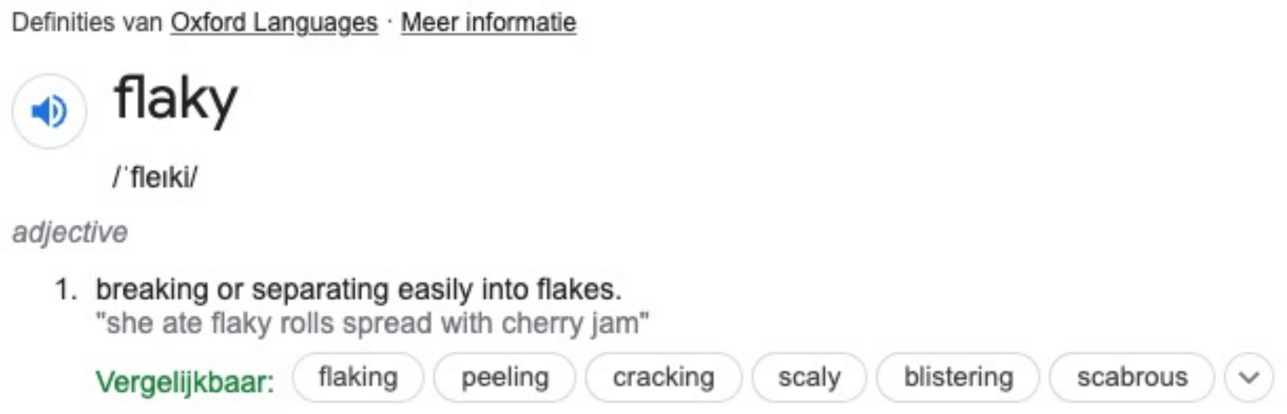
1. breaking or separating easily into flakes.
   "she ate flaky rolls spread with cherry jam"

Vergelijkbaar: flaking peeling cracking scaly blistering scabrous

- A test which is not 100% consistent is a **broken test**

Expensive, slow and <u>flaky</u> UI tests

- What is flaky?
  - **Breaking**
  - **Inconsistent**



Definities van Oxford Languages · Meer informatie

flaky

/ˈfleɪki/

adjective

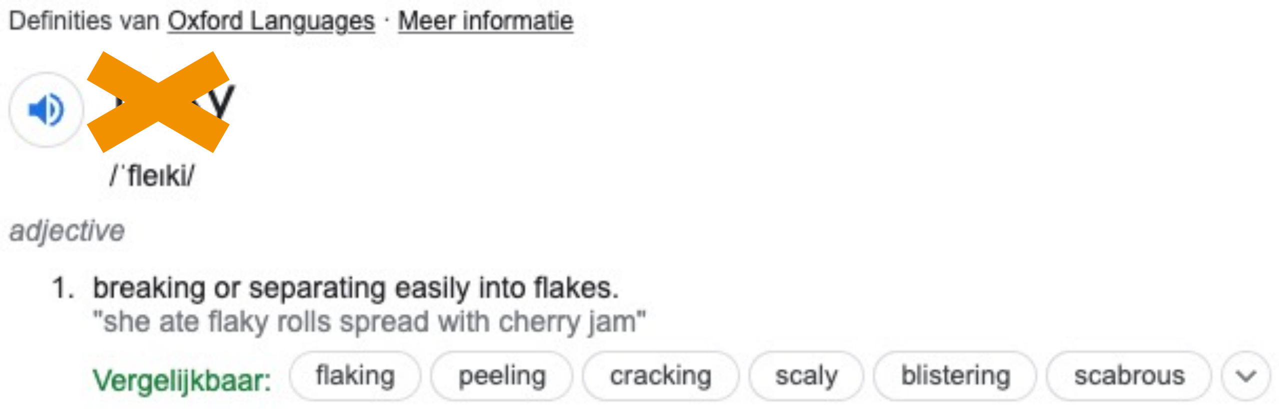
1. breaking or separating easily into flakes.
   "she ate flaky rolls spread with cherry jam"

- A Test which is not 100% consistent is a **broken test**

- There is **<u>always</u>** a reason why a test is not consistent

Expensive, slow and flaky UI tests

- UI == Software
- Browser == Software
  - **No opinion**
  - **0 and 1**
  - **Pure logic**



Definities van Oxford Languages · Meer informatie

/ˈfleɪki/

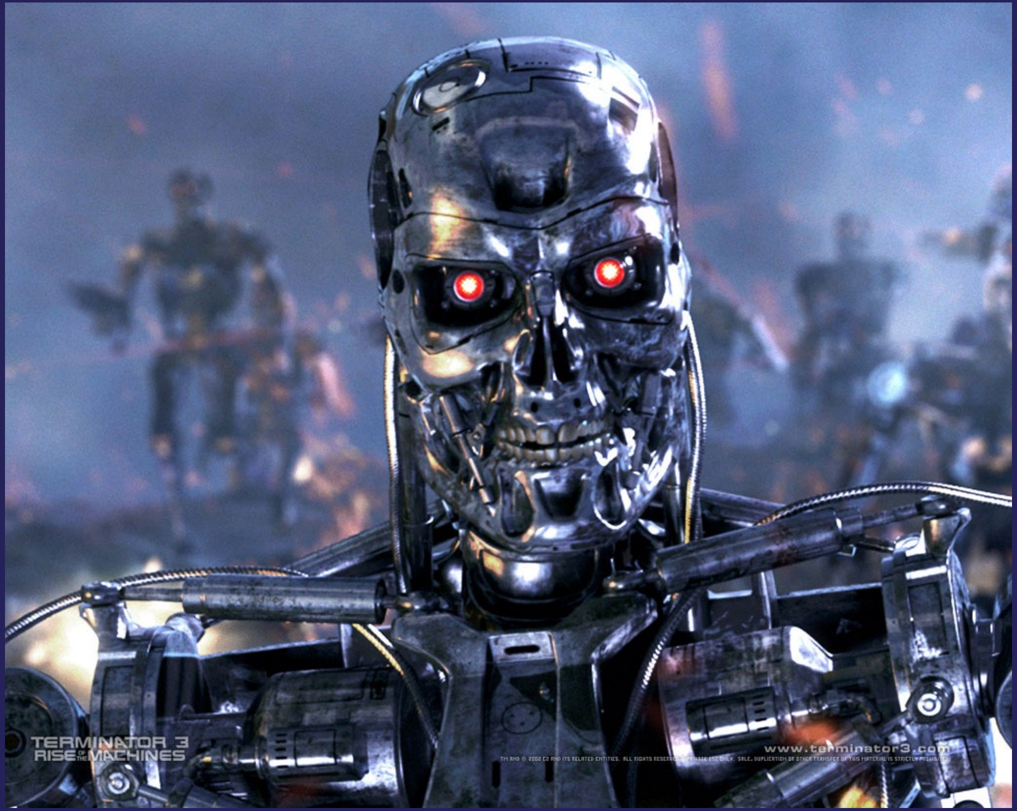
adjective

1. breaking or separating easily into flakes.
   "she ate flaky rolls spread with cherry jam"

   Vergelijkbaar: flaking | peeling | cracking | scaly | blistering | scabrous

Flaky means “I don’t know”

Issues can always be explained, except:

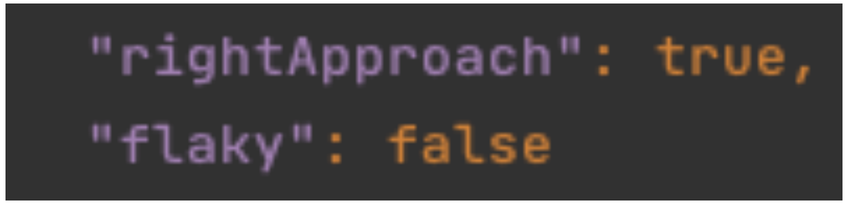


“Your automation effort will be expensive, slow and flaky!”

“Your automation effort will be expensive, ~~slow~~ and ~~flaky~~!”

Right approach to UI test automation

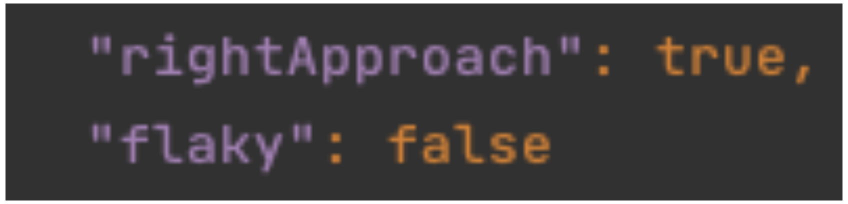
- Right/Proper/Correct, oh that’s easy ☺
  - **It is hard**
  - **It is a skill**



```
"rightApproach": true,
"flaky": false
```

Right approach to UI test automation

- Right/Proper/Correct, oh that’s easy ☺
  - **It is hard**
  - **It is a skill**

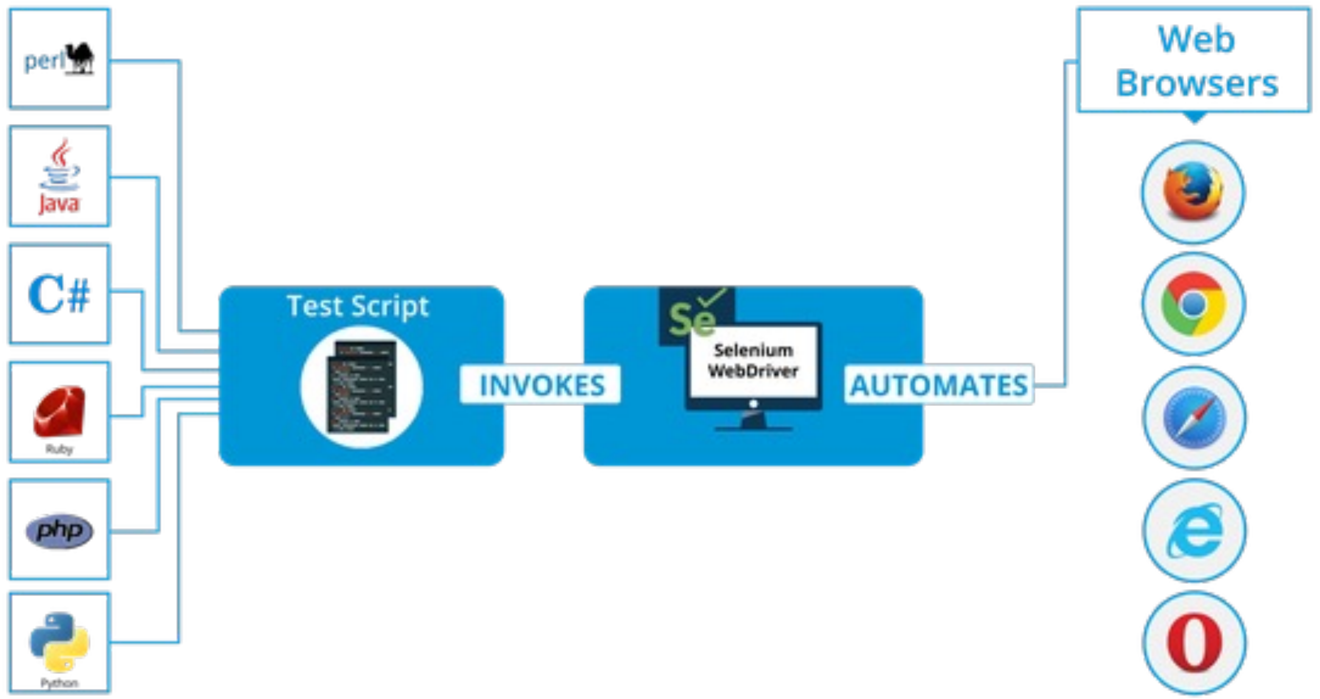


```
"rightApproach": true,
"flaky": false
```

- Understand what’s going on
  - **Mindset -> Automation the interaction with the UI -> Mimicking an end user**
  - **High level -> Writing code to create an automated UI test**
  - **Low level ->**

Right approach to UI automation

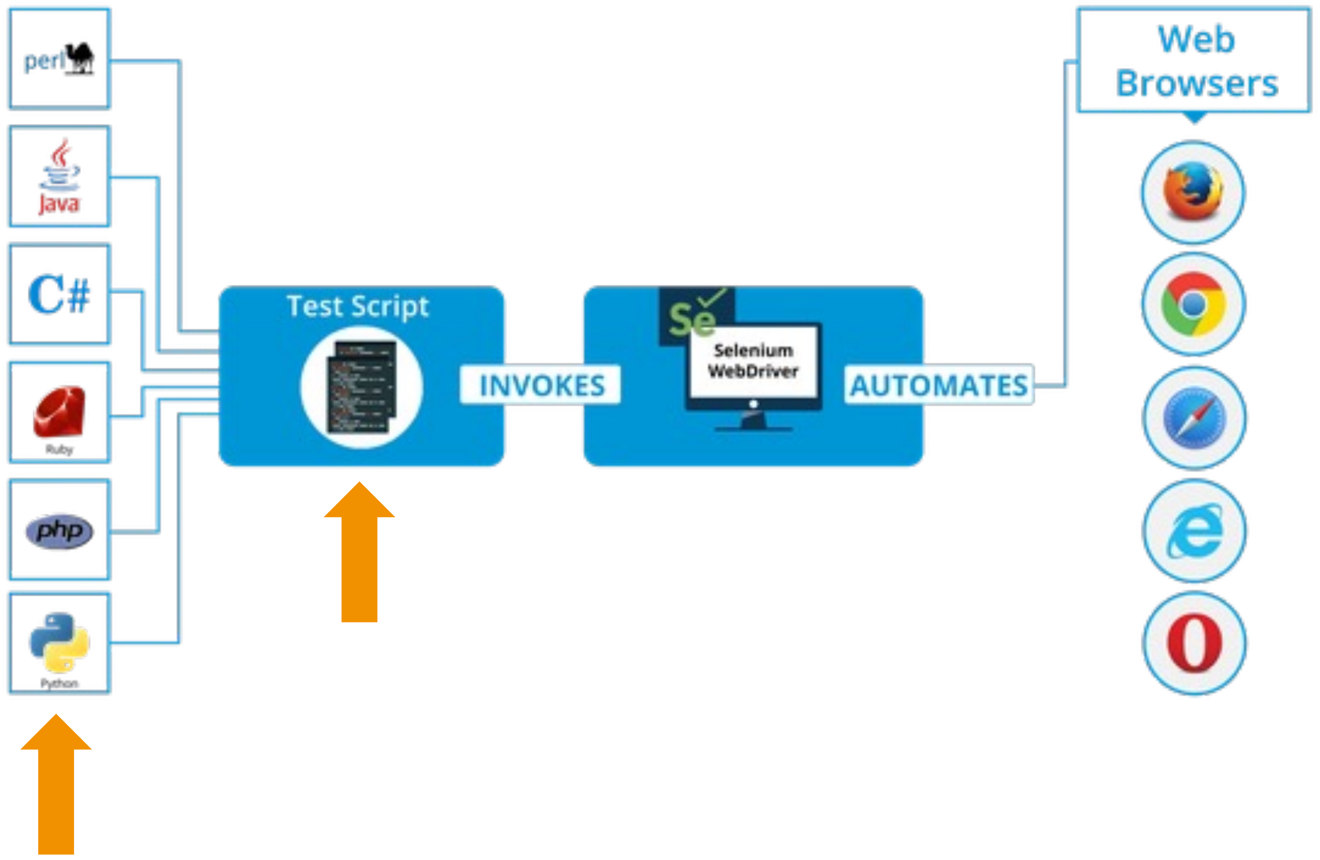
- Low level explained



Source: danielgold.net

Right approach to UI automation

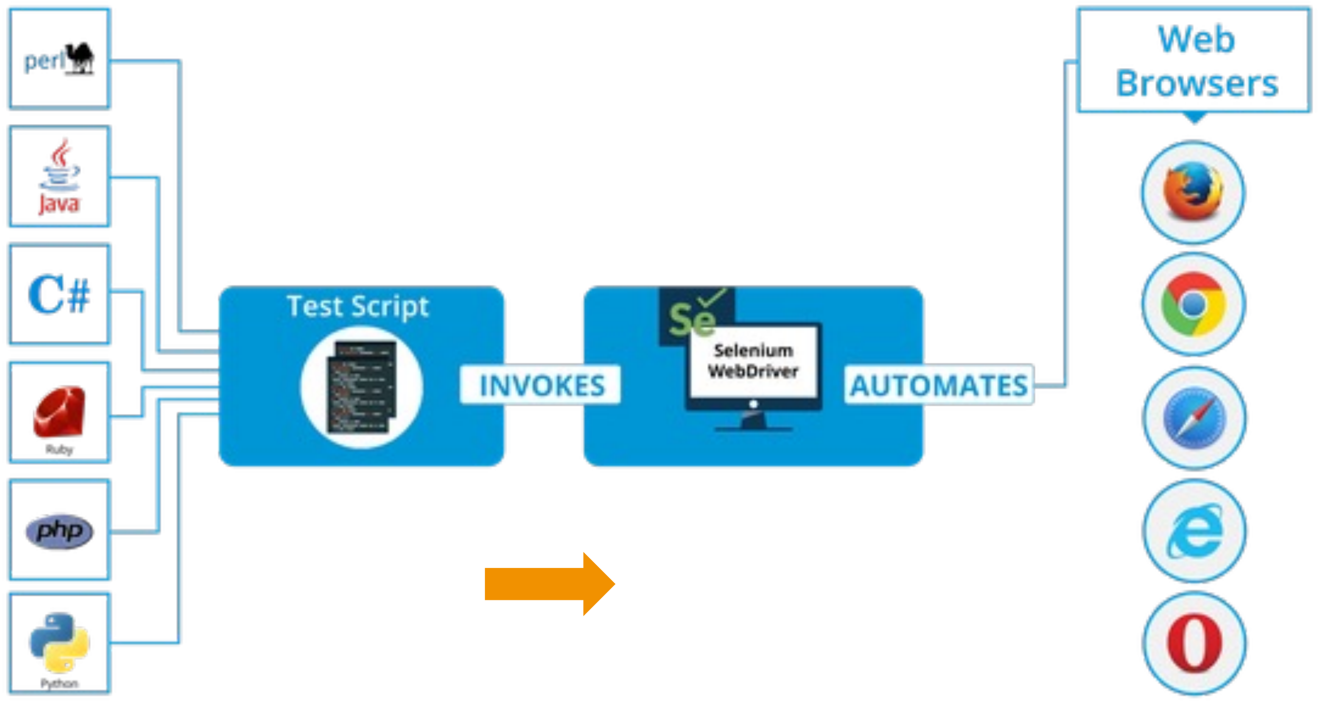
- Our written Selenium WebDriver code..



Source: danielgold.net

Right approach to UI automation

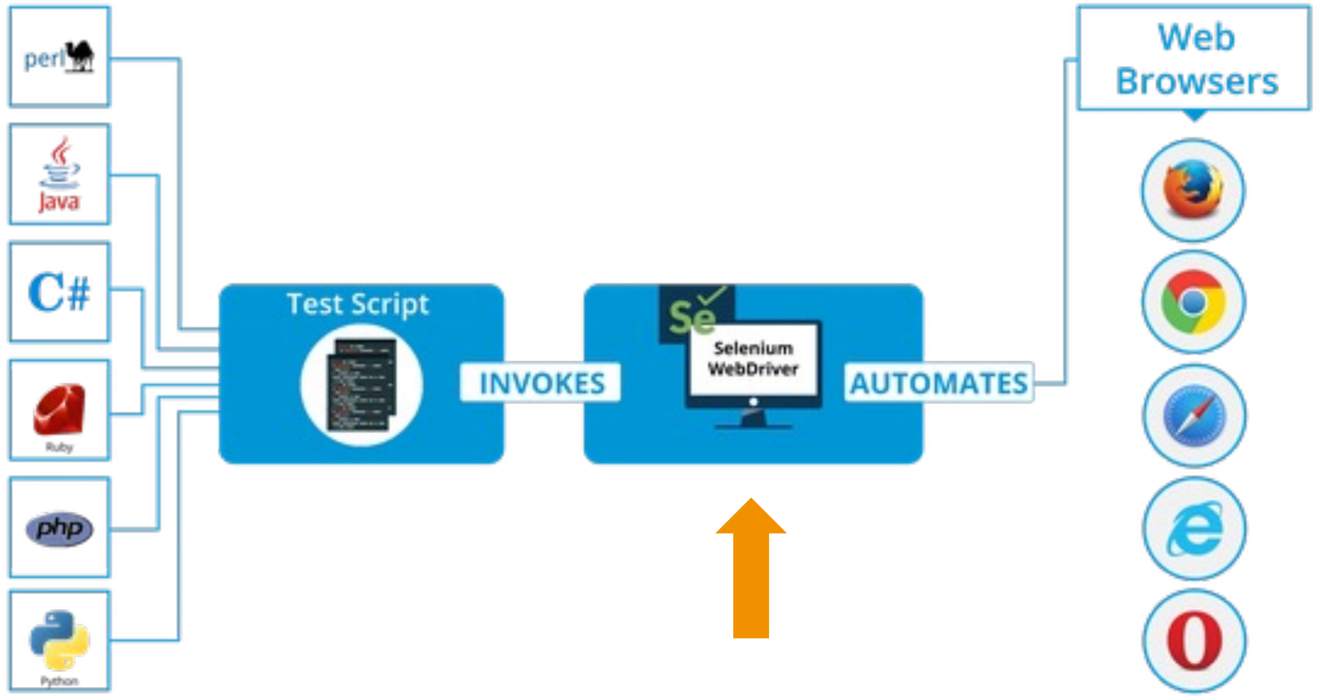
- ..starts a session with the chromedriver.exe on port 17494



Source: danielgold.net

Right approach to UI automation

- Chromedriver.exe acts as a server. Its role is to interact with the browser.



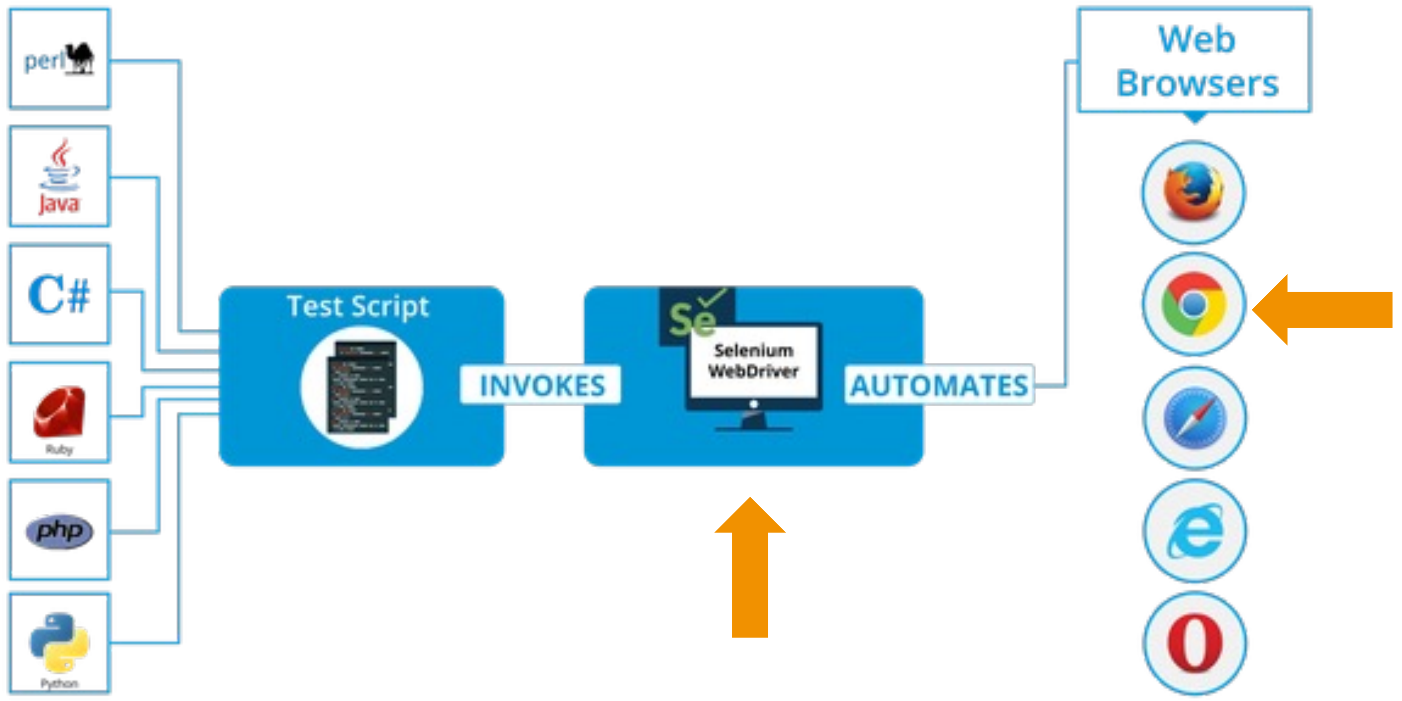
Building IT Quality

Source: danielgold.net

Right approach to UI automation

VALORI

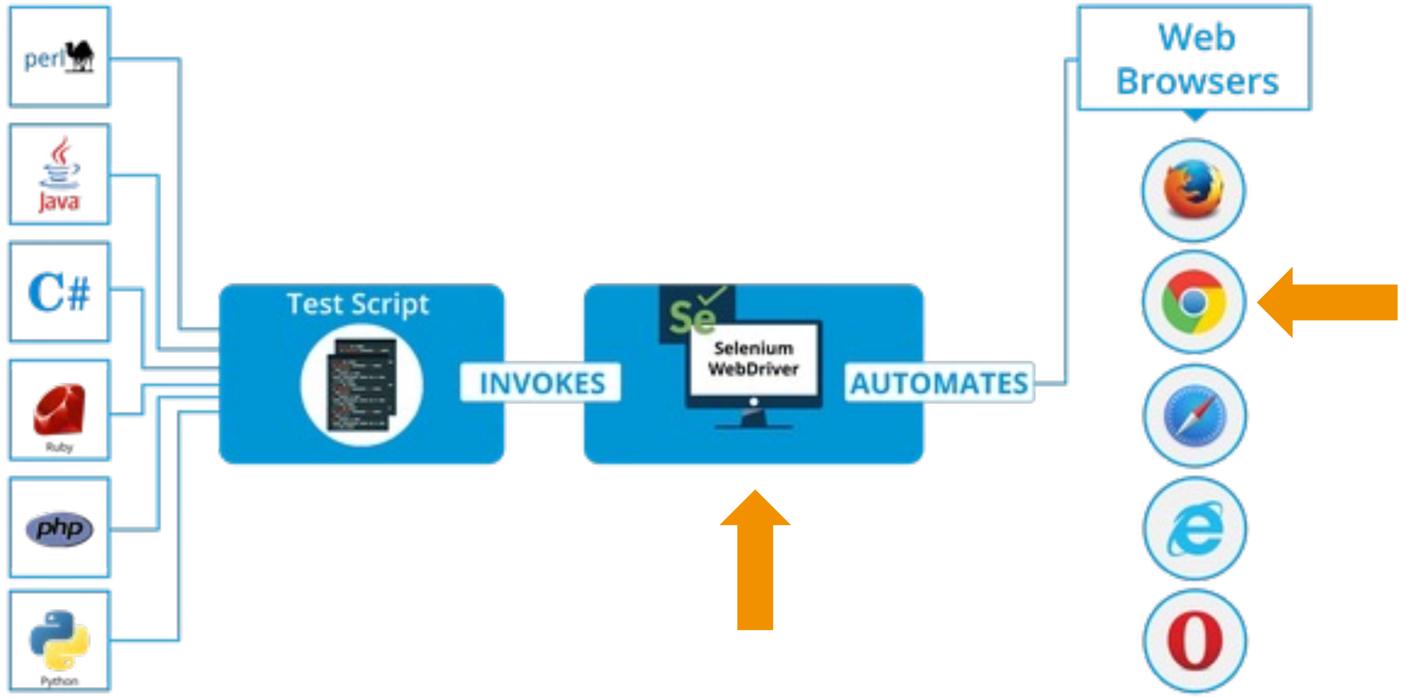
- Made by Google themselves!



Source: danielgold.net

Right approach to UI automation

- Google knows their own product best, the server should be robust, reliable and stable. Should be ☺

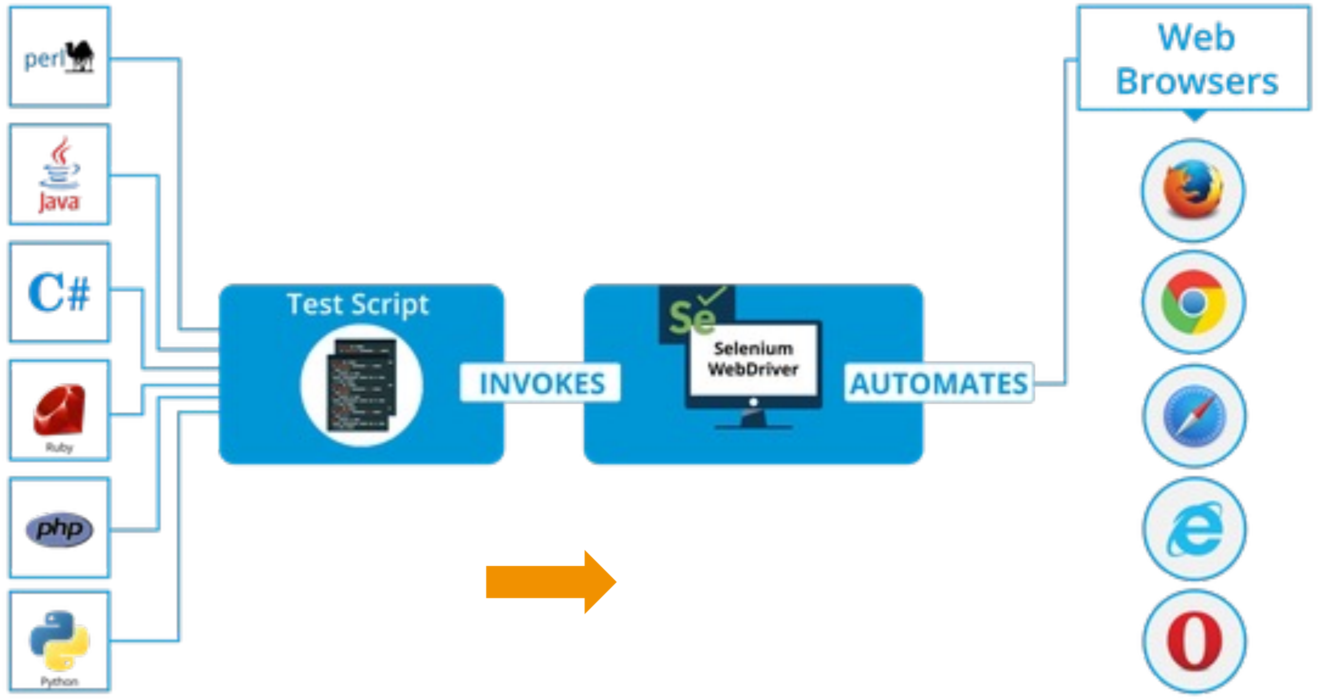


Source: danielgold.net

Right approach to UI automation

VALORI

- Chromedriver.exe acts as a server and accepts REST commands. Specific WebDriver protocols

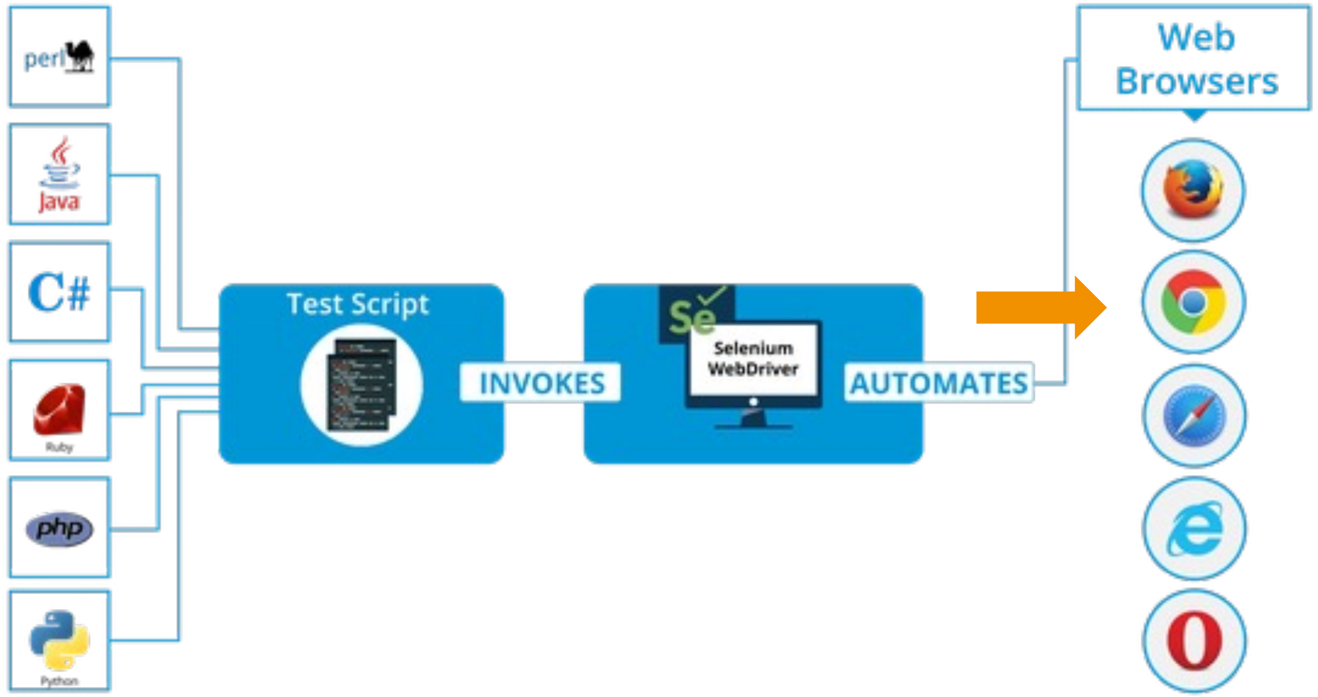


Building IT Quality

Source: danielgold.net

Right approach to UI automation

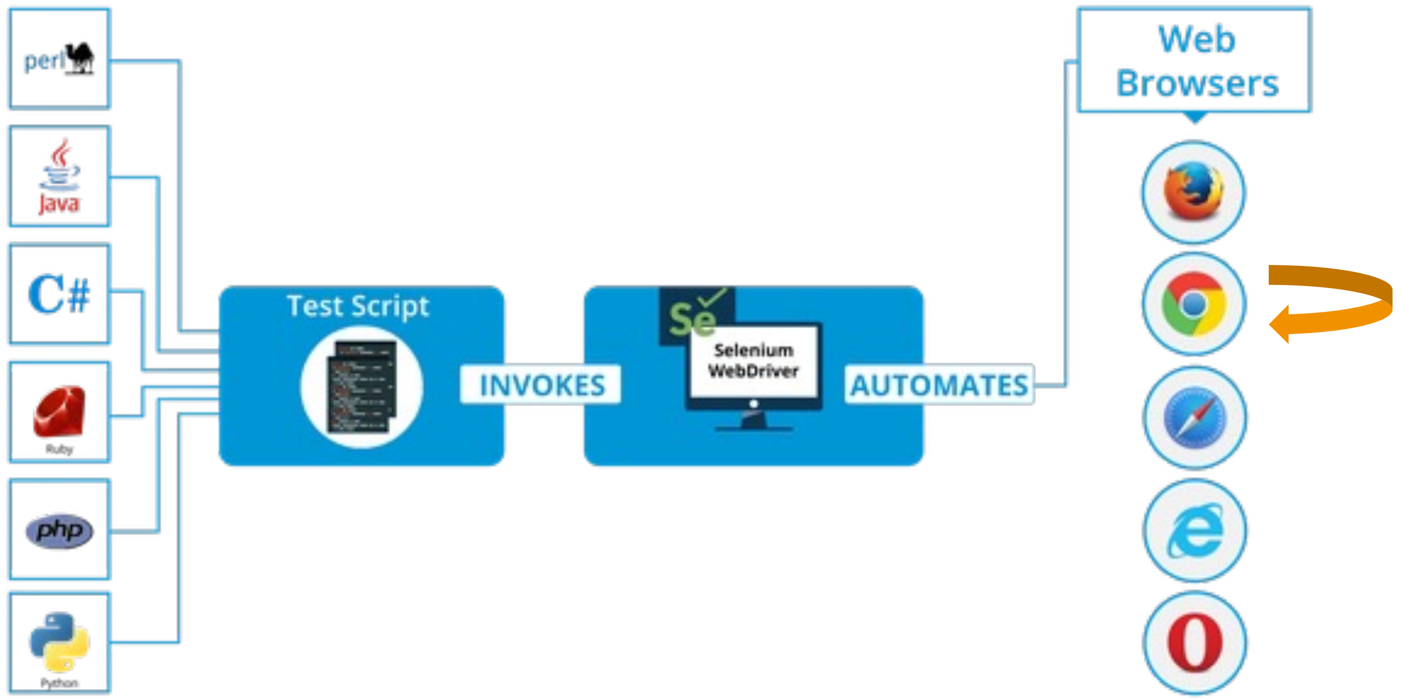
- Chromedriver.exe interacts with the browser via the Remote Debugger Protocol



Source: danielgold.net

Right approach to UI automation

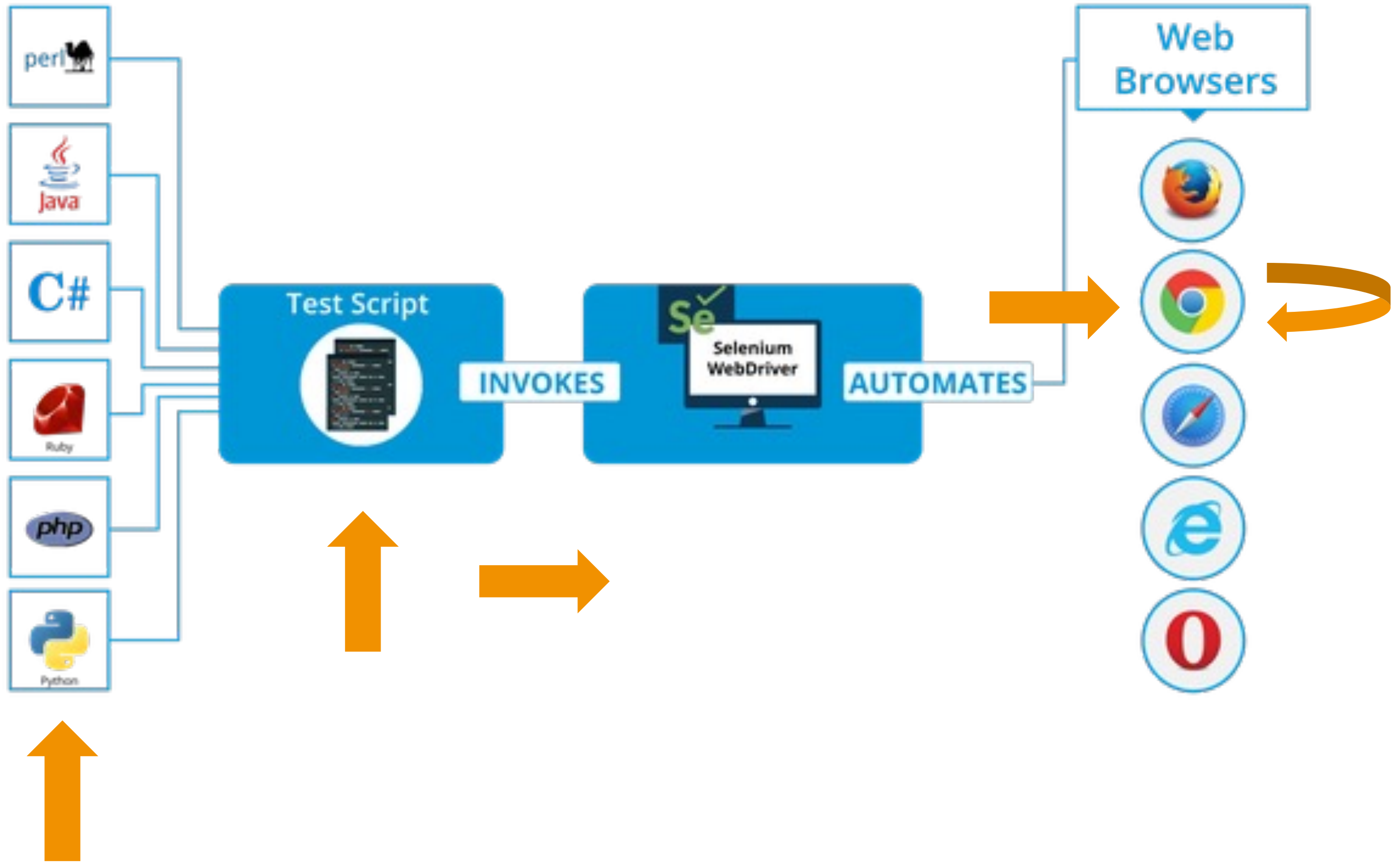
- Lastly the RDP translates the API calls so chromium JS code can be executed.



Source: danielgold.net

Right approach to UI automation

- Test code -> ChromeDriver.exe -> RDP -> Chromium JS code



Source: danielgold.net

Right approach to UI automation

VALORI

- Playwright talks directly to the Chrome Debugger tool

Right approach to UI automation

- Playwright talks directly to the Chrome Debugger tool

- Cypress
  - Cypress bundles jQuery and exposes many of its DOM traversal methods to you
  - Cypress loads its test code in the test iframe,
  - While the application is running in another iframe in the same browser tab
  - Test code direct access to the application and most of the browser APIs
  - JavaScript in the same browser tab

Right approach to UI automation

VALORI

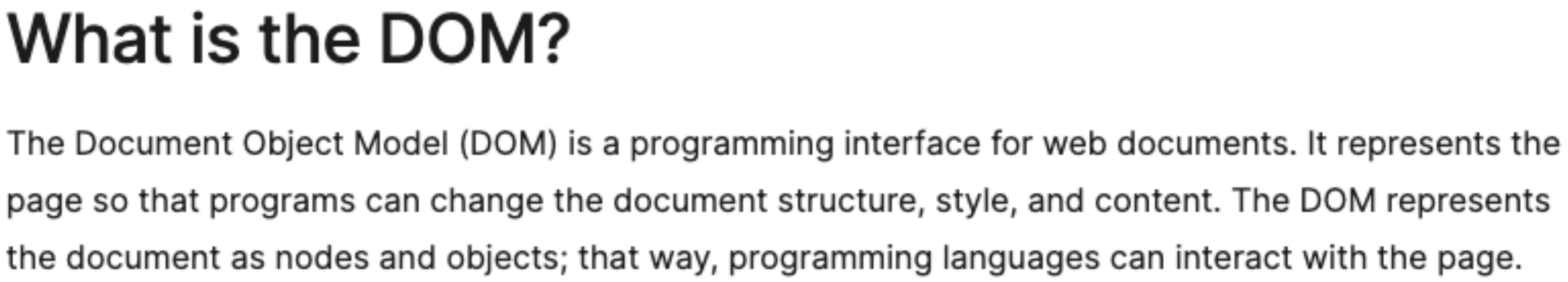
- Playwright talks directly to the Chrome Debugger tool

- Cypress
  - Cypress bundles jQuery and exposes many of its DOM traversal methods to you
  - Cypress loads its test code in the test iframe,
  - While the application is running in another iframe in the same browser tab
  - Test code direct access to the application and most of the browser APIs
  - JavaScript in the same browser tab

- Low code tooling
  - **Proprietary, ask them ☺**

Right approach to UI automation

- Not looking at renders from the UI
- Talking directly to the browser or the next best thing
- All about the DOM
  - **Document Object Model**



What is the DOM?

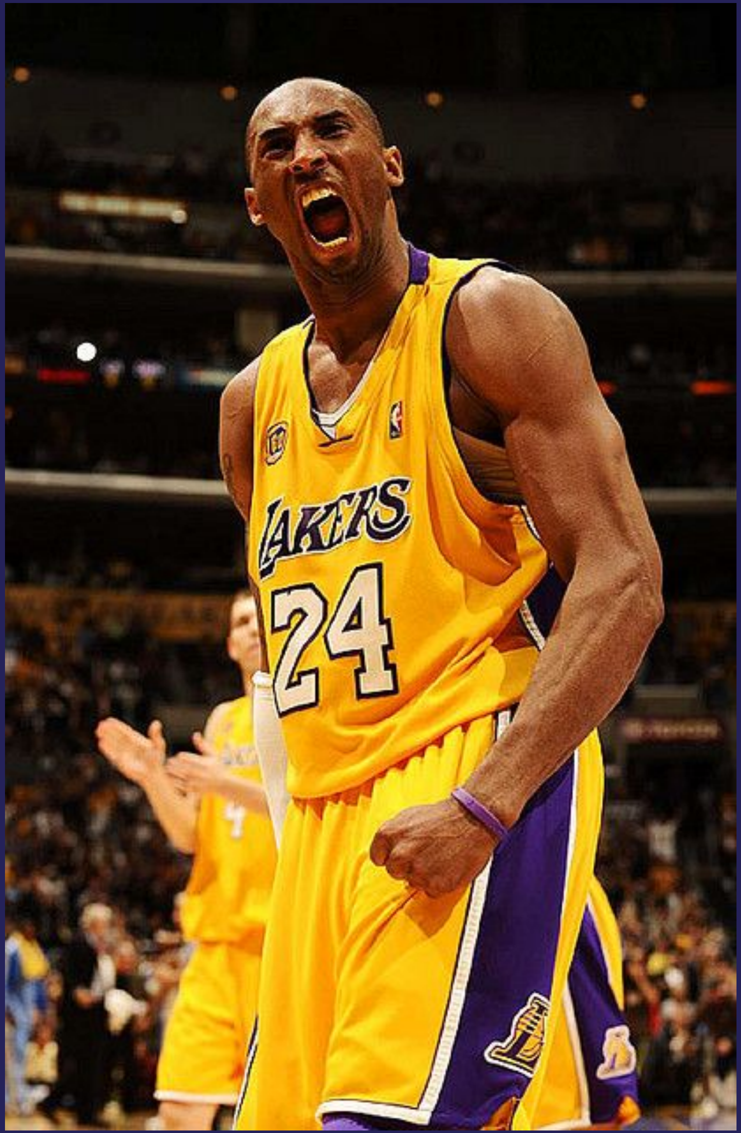
The Document Object Model (DOM) is a programming interface for web documents. It represents the page so that programs can change the document structure, style, and content. The DOM represents the document as nodes and objects; that way, programming languages can interact with the page.

Right approach to UI automation

- Benefits of this approach?
- How will me test case use this approach?

Fast

Vision



Mentality

Polaroid Spectra System.
3:51
Gatorade

Super user

Budweiser
Gatorade
JORDAN
97
PLR F PTS
WILKINS
195
PLR F PTS
Gatorade
SLAM DUNK CHAMPIONSHIP
Winston
Coca-Cola
Gatorade
Slam-Dunk Championship

Super user

Super user

Super user

Right approach to UI automation

- Super user, super problems

Right approach to UI automation

- Super user, super problems
- Sometimes think like a regular user

Right approach to UI automation

- Super user, super problems

- Sometimes think like a regular user

- Wait, wait, wait, wait
  - **Because of the speed**
  - **Because of the vision**
  - **“Give the UI some time to breath”**
  - **The DOM must wait for the Front End to catch up (milliseconds!)**

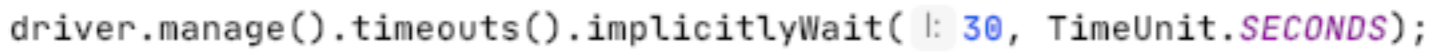
Right approach to UI automation

- Build a wait strategy

A WAIT STRATEGY

Right approach to UI automation

- Build a wait strategy



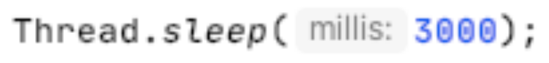
```
driver.manage().timeouts().implicitlyWait( l: 30, TimeUnit.SECONDS);
```

By implicitly waiting, WebDriver polls the DOM for a certain duration when trying to find *any* element.

“Wait for everything please”

Right approach to UI automation

- Build a wait strategy



```
Thread.sleep( millis: 3000);
```

Hard coded waits, native code waits

“Please block all calls”

Right approach to UI automation

- Build a wait strategy
- Zero in on your element

Right approach to UI automation

- Build a wait strategy
- Zero in on your element
- Wait for it to be…
  - **Visible**
  - **Existing**
  - **Interactable**
  - **Clickable**
  - **Focused**
  - **Selected**
  - **Stale**
  - **Invisible**

Right approach to UI automation

VALORI

- Build a wait strategy
- Zero in on your element
- Wait for it to be…
  - **Visible**
  - **Existing**
  - **Interactable**
  - **Clickable**
  - **Focused**
  - **Selected**
  - **Stale**
  - **Invisible**

- Then interact

Right approach to UI automation

VALORI

- Build a wait strategy
- Zero in on your element
- Wait for it to be…
  - **Visible**
  - **Existing**
  - **Interactable**
  - **Clickable**
  - **Focused**
  - **Selected**
  - **Stale**
  - **Invisible**

1. User your imagination!
2. The documentation
3. Think: “I need help out my super user”

- Then interact

Right approach to UI automation

- Focus is important

Right approach to UI automation

- Focus is important

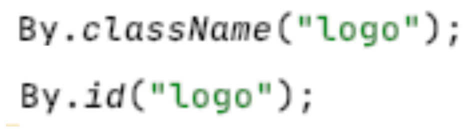
- Locating an element on the page
  - **Locators!**

Right approach to UI automation

- Focus is important

- Locating an element on the page
  - **Locators!**

- Build your own locators, always
  - **Don’t use the prebuild identifiers, use XPath or CSS locators**



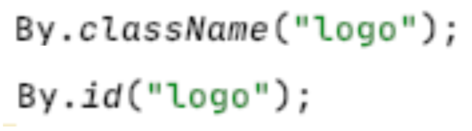
```
By.className("logo");
By.id("logo");
```

Right approach to UI automation

- Focus is important

- Locating an element on the page
  - **Locators!**

- Build your own locators, always
  - **Don’t use the prebuild identifiers, use XPath or CSS locators**



```
By.className("logo");
By.id("logo");
```

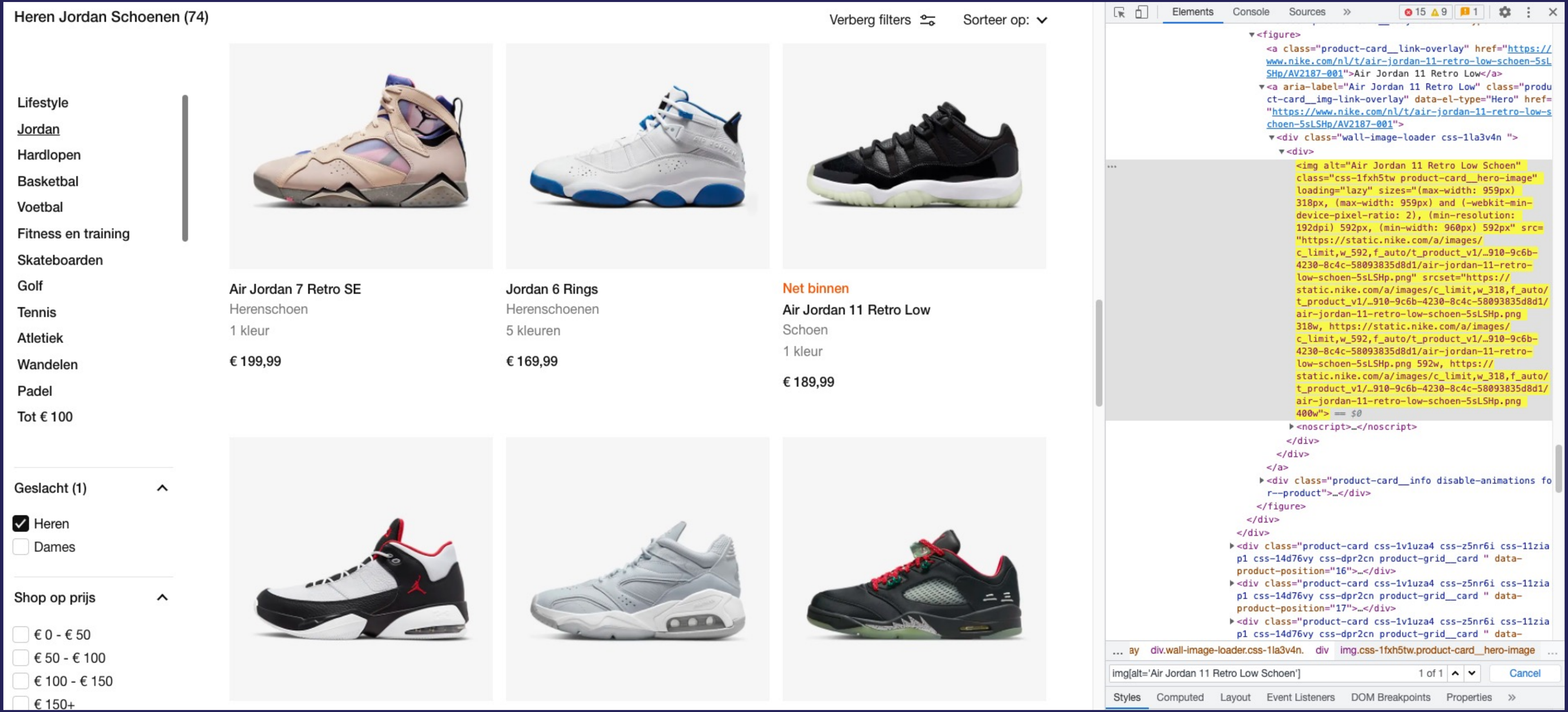
- Understand not only the DOM but the cohesion of the DOM, the structure

Right approach to UI automation

- Focus is important
- Locating an element on the page
  - **Locators!**
- Build your own locators, always
  - **Don’t use the prebuild identifiers, use XPath or CSS locators**

```
By.className("logo");
By.id("logo");
```

- Understand not only the DOM but the cohesion of the DOM, the structure
- Your locator must be unique



Heren Jordan Schoenen (74)

Verberg filters   Sorteer op:

- Lifestyle
- Jordan
- Hardlopen
- Basketbal
- Voetbal
- Fitness en training
- Skateboarden
- Golf
- Tennis
- Atletiek
- Wandelen
- Padel
- Tot € 100

Geslacht (1)

- [✓] Heren
- [ ] Dames

Shop op prijs

- [ ] € 0 - € 50
- [ ] € 50 - € 100
- [ ] € 100 - € 150
- [ ] € 150+

Air Jordan 7 Retro SE
Herenschoen
1 kleur
€ 199,99

Jordan 6 Rings
Herenschoenen
5 kleuren
€ 169,99

Net binnen
Air Jordan 11 Retro Low
Schoen
1 kleur
€ 189,99

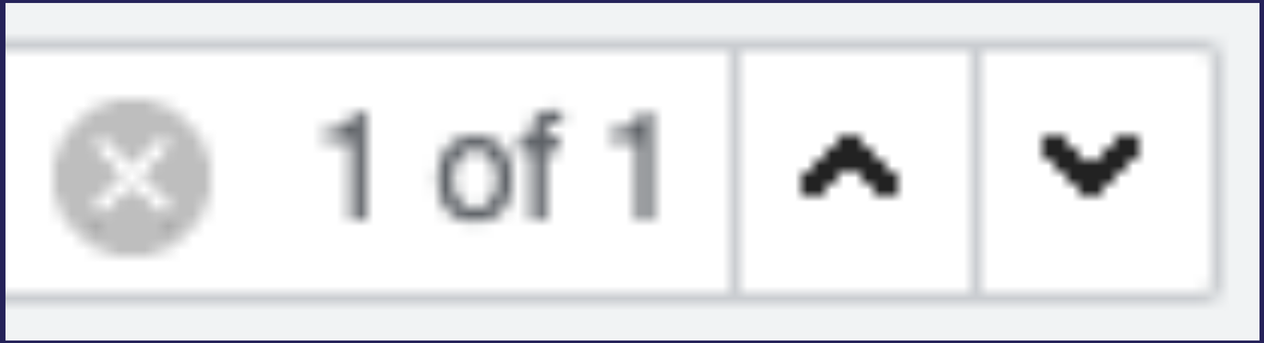
Elements   Console   Sources   »   15 9 1

```
▼<figure>
  <a class="product-card__link-overlay" href="https://
  www.nike.com/nl/t/air-jordan-11-retro-low-schoen-5sL
  SHp/AV2187-001">Air Jordan 11 Retro Low</a>
 ▼<a aria-label="Air Jordan 11 Retro Low" class="produ
  ct-card__img-link-overlay" data-el-type="Hero" href=
  "https://www.nike.com/nl/t/air-jordan-11-retro-low-s
  choen-5sLSHp/AV2187-001">
   ▼<div class="wall-image-loader css-1la3v4n ">
     ▼<div>
         <img alt="Air Jordan 11 Retro Low Schoen"
         class="css-1fxh5tw product-card__hero-image"
         loading="lazy" sizes="(max-width: 959px)
         318px, (max-width: 959px) and (-webkit-min-
         device-pixel-ratio: 2), (min-resolution:
         192dpi) 592px, (min-width: 960px) 592px" src=
         "https://static.nike.com/a/images/
         c_limit,w_592,f_auto/t_product_v1/…910-9c6b-
         4230-8c4c-58093835d8d1/air-jordan-11-retro-
         low-schoen-5sLSHp.png" srcset="https://
         static.nike.com/a/images/c_limit,w_318,f_auto/
         t_product_v1/…910-9c6b-4230-8c4c-58093835d8d1/
         air-jordan-11-retro-low-schoen-5sLSHp.png
         318w, https://static.nike.com/a/images/
         c_limit,w_592,f_auto/t_product_v1/…910-9c6b-
         4230-8c4c-58093835d8d1/air-jordan-11-retro-
         low-schoen-5sLSHp.png 592w, https://
         static.nike.com/a/images/c_limit,w_318,f_auto/
         t_product_v1/…910-9c6b-4230-8c4c-58093835d8d1/
         air-jordan-11-retro-low-schoen-5sLSHp.png
         400w"> == $0
       ▶<noscript>…</noscript>
       </div>
     </div>
   </a>
 ▶<div class="product-card__info disable-animations fo
  r--product">…</div>
 </figure>
</div>
</div>
▶<div class="product-card css-1v1uza4 css-z5nr6i css-11zia
 p1 css-14d76vy css-dpr2cn product-grid__card " data-
 product-position="16">…</div>
▶<div class="product-card css-1v1uza4 css-z5nr6i css-11zia
 p1 css-14d76vy css-dpr2cn product-grid__card " data-
 product-position="17">…</div>
▶<div class="product-card css-1v1uza4 css-z5nr6i css-11zia
 p1 css-14d76vy css-dpr2cn product-grid__card " data-
```

… ay div.wall-image-loader.css-1la3v4n. div img.css-1fxh5tw.product-card__hero-image …

img[alt='Air Jordan 11 Retro Low Schoen']   1 of 1   Cancel

Styles   Computed   Layout   Event Listeners   DOM Breakpoints   Properties   »



All about this!

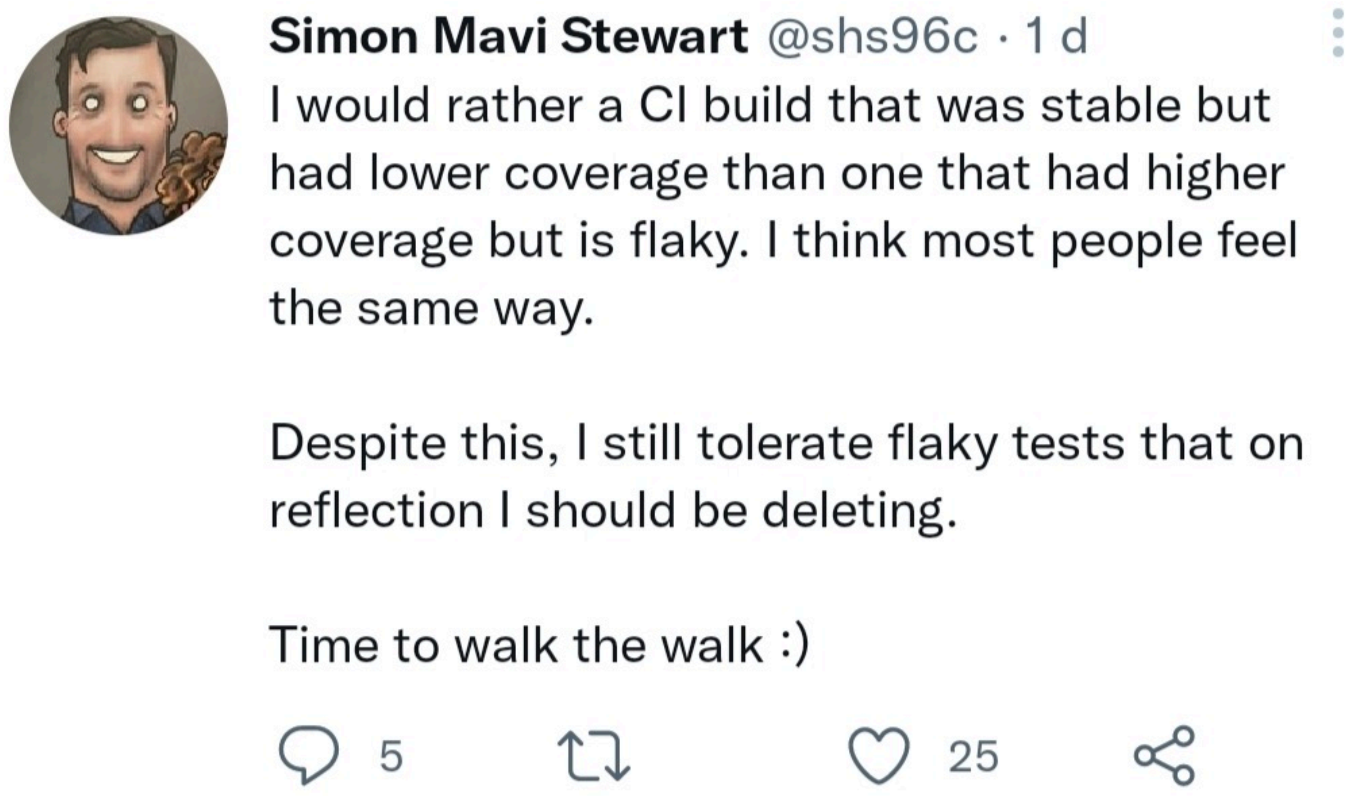
Right approach to UI automation

- Concurrent runs
  - **Can your application handle that?**
- Your runs must be consisted, every time
  - **Test data ☹**
- Take it to the next level
  - **BE short cuts**
- Keep trying
  - **It is hard**
  - **It is a skill**

But what if the test remains “flaky” ☹

But what if the test remains “flaky” ☹

But what if the test remains “flaky” ☹



Simon Mavi Stewart @shs96c · 1 d

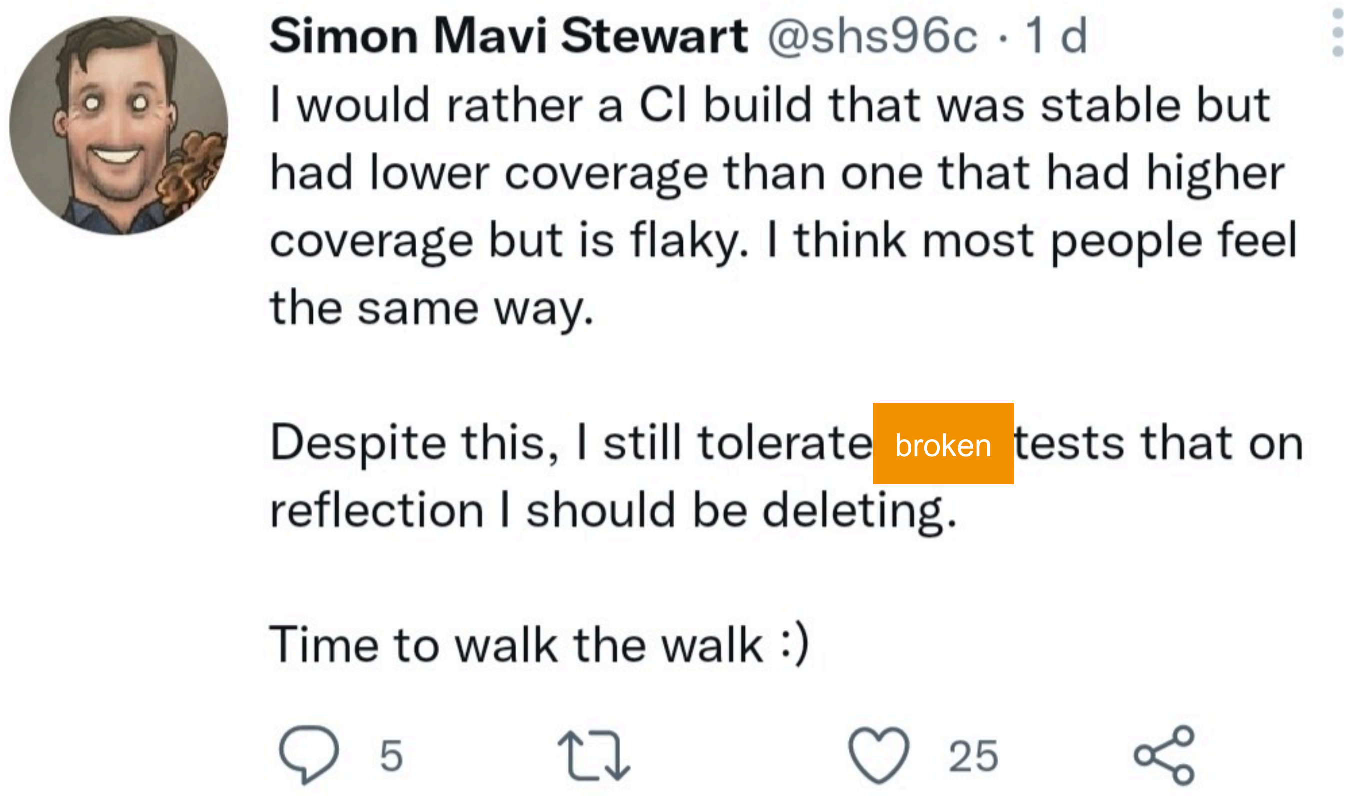
I would rather a CI build that was stable but had lower coverage than one that had higher coverage but is flaky. I think most people feel the same way.

Despite this, I still tolerate flaky tests that on reflection I should be deleting.

Time to walk the walk :)

5 25

But what if the test remains “flaky” ☹


Simon Mavi Stewart @shs96c · 1 d

I would rather a CI build that was stable but had lower coverage than one that had higher coverage but is flaky. I think most people feel the same way.

Despite this, I still tolerate broken tests that on reflection I should be deleting.

Time to walk the walk :)

5 25

Embrace your super user

He will not let you down

Bedankt voor je aandacht!

Thank you for listening!

Contactgegevens:

MichelLalmohamed@Valori.nl

@michelamin47