

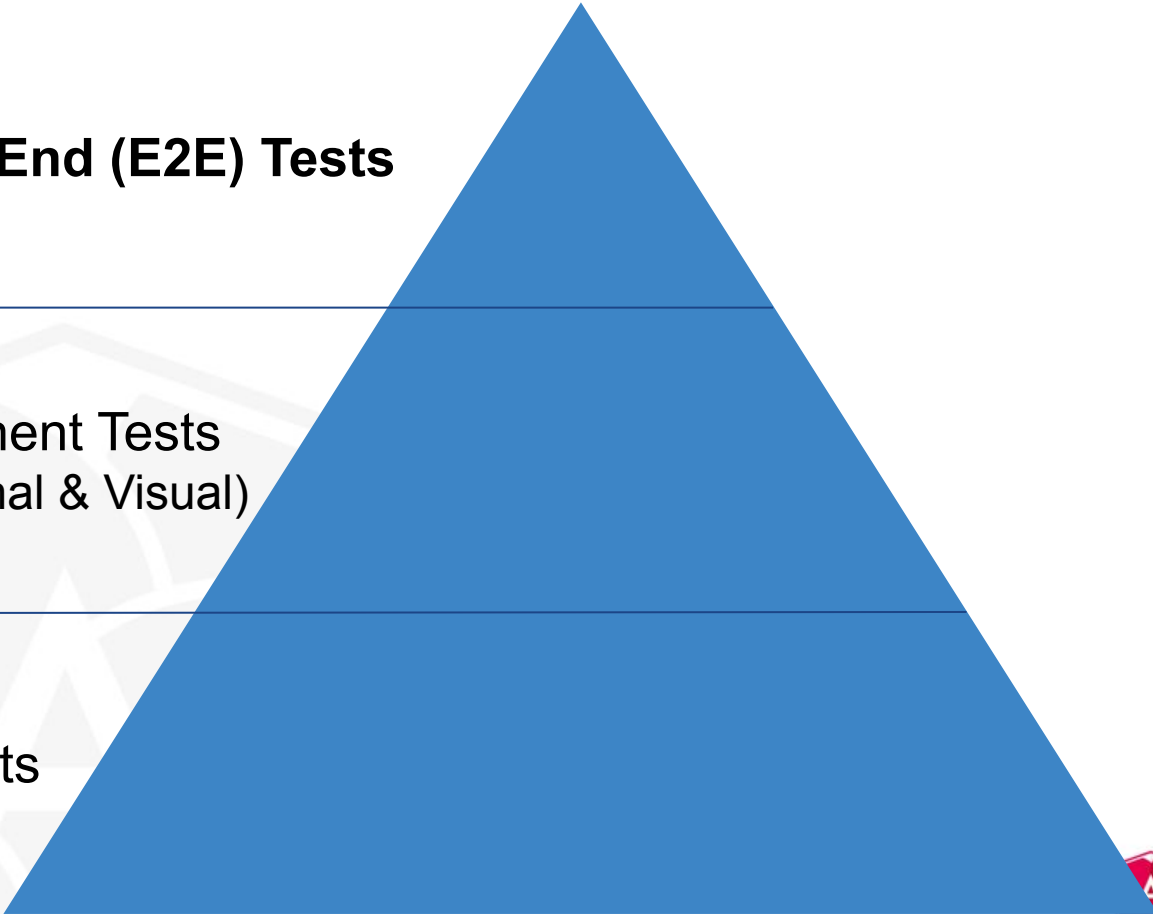


ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

Angular Testing

1 - E2E Basics

End-to-End (E2E) Tests

A blue pyramid diagram divided into three horizontal sections by two thin blue lines. The top section is the smallest, the middle section is medium-sized, and the bottom section is the largest. The text labels are positioned to the left of the pyramid, aligned with their respective sections.

**Component Tests
(Functional & Visual)**

Unit Tests



ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE



Selenium RC

2005



Selenium 2

2010

CDP

2015



Cypress



Playwright

2020

2022



ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

2 Core Technologies for E2E Frameworks

1. WebDriver
2. Chrome DevTool Protocol (CDP)



ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

WebDriver

- W3C Standard
- Cross-Browser Support
- Known to be flaky
- WebDriver BiDi as successor
- Popular E2E Tools
 - Protractor
 - Selenium
 - NightWatch
 - WebDriverIO



CDP (Chrome DevTools Protocol)

- Debugging Tool for Chromium-based Browsers
- Puppeteer as primary library
- Playwright incl. Safari
- Much more stable
- Constrained to CDP-Browsers



Cross Browser
Support

Reliability

WebDriver



CDP
(Chrome DevTools Protocol)



Native



ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

Cross Browser
Support

Reliability

WebDriver

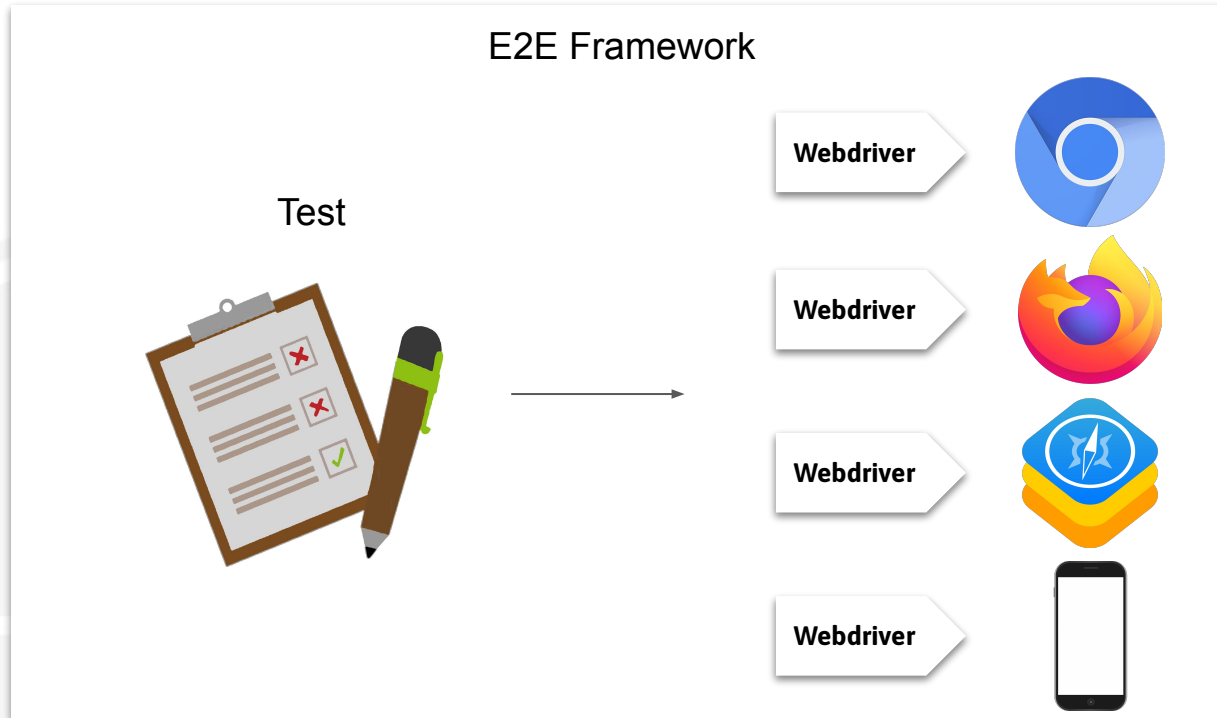
CDP
(Chrome DevTools Protocol)

Native

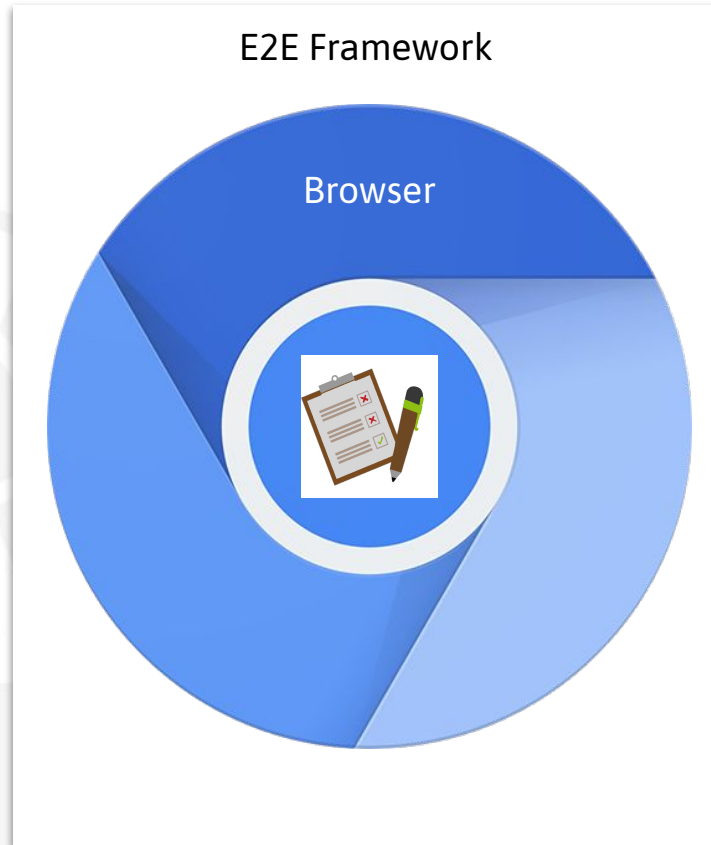


ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

Selenium: Outside browser testing



Cypress: Inside browser testing



@playwright/test vs cypress vs nightwatch vs testcafe vs webdriverio

Enter an npm package...

@playwright/test x

cypress x

nightwatch x

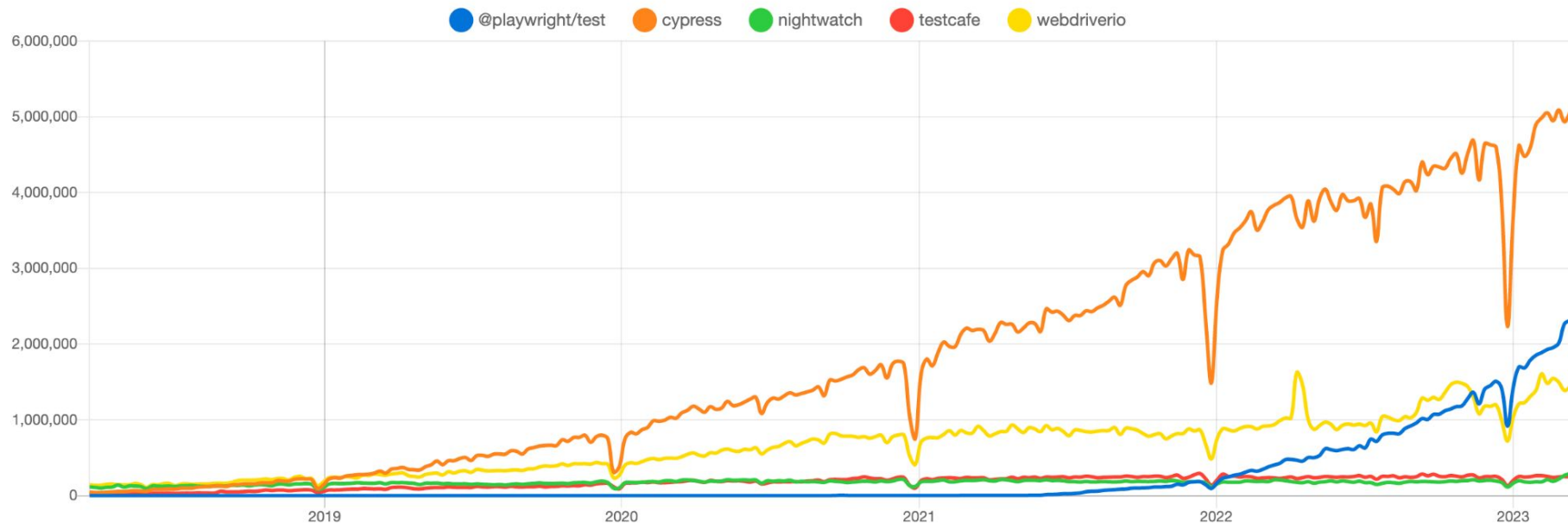
testcafe x

webdriverio x

+ puppeteer

+ playwright

Downloads in past 5 Years ▾



Cypress

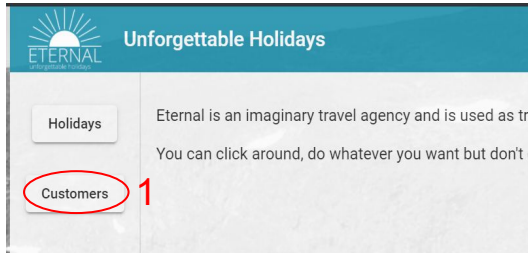
- Node.js Application
- Great Developer Experience
- Good Documentation
- Easy Setup
- Internal "IDE"
- CI Features like Videorecording and Screenshots



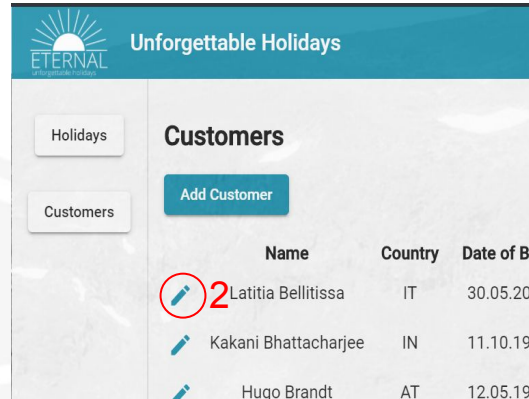
ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

The Await Feature "Actionability"





1. Select Customers



2. Select Customer



3. Edit Firstname



ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE



Unforgettable Holidays

Holidays

Customers

Firstname *

Latitia

Name *

Bellitissa

Country *

Italy

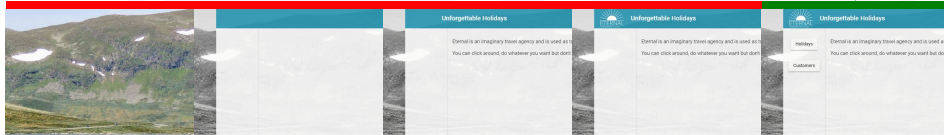
Birthdate *

30.5.2002

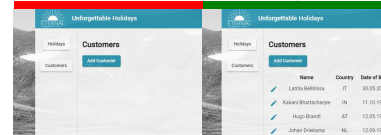


Non-Waiting Style

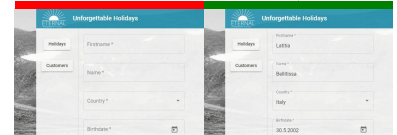
1. Select Customers



2. Select Customer

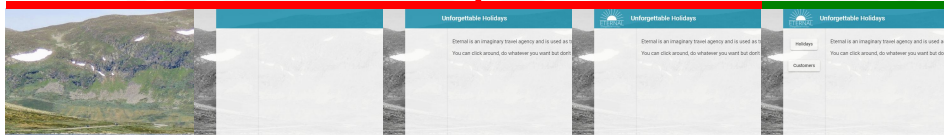


3. Edit Firstname

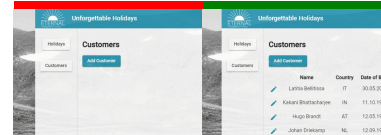


Non-Waiting Style

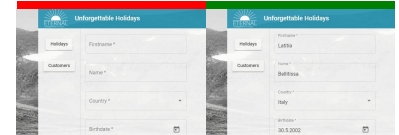
1. Select Customers



2. Select Customer

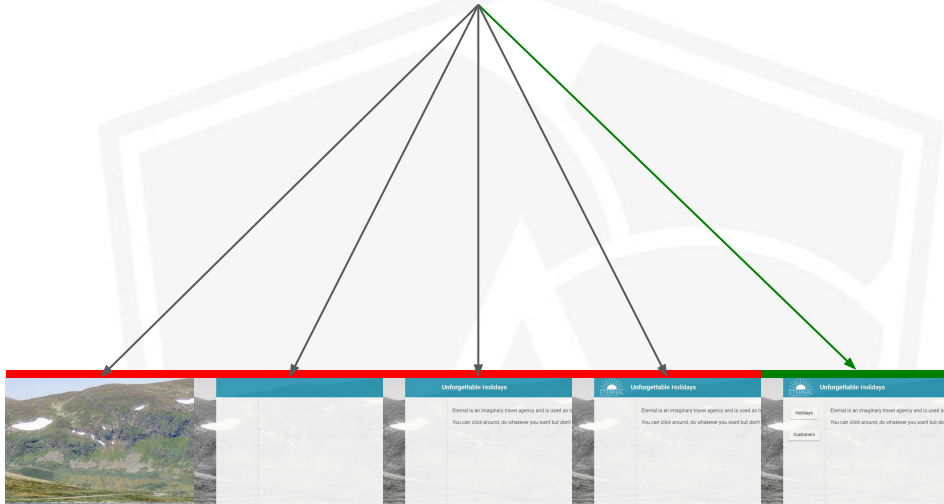


3. Edit Firstname

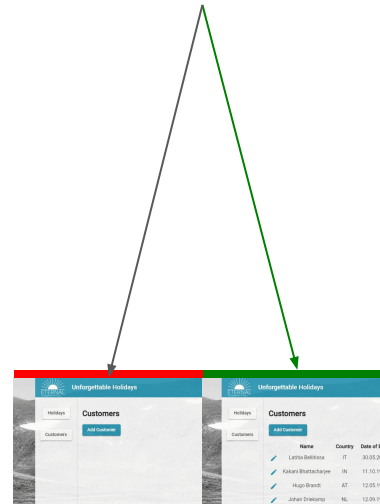


Waiting Style

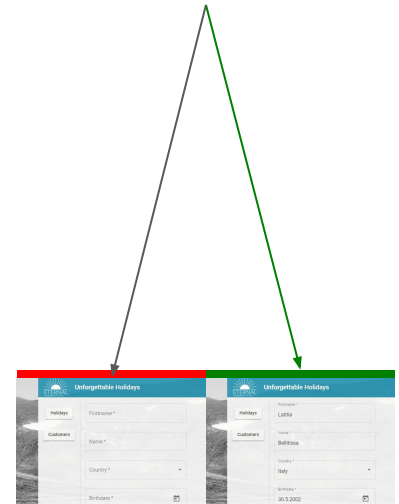
1. Select Customers



2. Select Customer



3. Edit Firstname



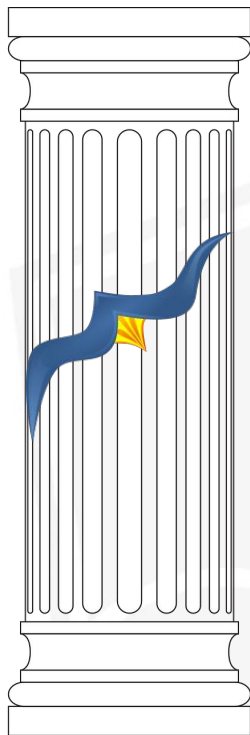
An E2E Test in Cypress

```
it('should rename Latitia to Laetitia', () => {  
  
  cy.visit('');  
  
  cy.get('[data-testid=btn-customers]').click();  
  
  cy.get('[data-testid=row-customer]')  
    .contains('Latitia Bellitissa')  
    .siblings('[data-testid=btn-edit]')  
    .click();  
  
  cy.get('[data-testid=inp-firstname]').clear().type('Laetitia');  
  
  cy.get('[data-testid=btn-submit]').click();  
  
  cy.get('[data-testid=row-customer]').contains('Bellitissa')  
    .should('have.text', 'Laetitia Bellitissa');  
  
});
```



Pillars

Bluebird



Mocha & Chai



jQuery



Search specs +

Specs ✓ 1 ✗ -- ↺ ↻

src / e2e

a11y.cy.ts

customers.cy.ts

holidays.cy.ts

init.cy.ts

misc.cy.ts

init.cy.ts 593ms

✓ should not work

▼ TEST BODY

1 visit

http://localhost:4200/

div.screenshot-height-container 553.03 x 365

Unforgettable Holidays

Holidays

Welcome to Eternal

Eternal is an imaginary travel agency and is used as training application for Angular developers.

You can click around, do whatever you want but don't expect to be able to book a real holiday 🤖

Customers

Newsletter

Settings

Mock Customers

Mock Holidays

Sign In

Elements

Console

Sources

Network

Performance

Memory

Application

Security

Lighthouse

Recorder

Performance insights

```
</div>
</div>
</div>
</div>
<div data-cy="panel2ResizeHandle" class="cursor-ew-resize h-full top-0 -right-6px w-10px z-30 absolute"></div>
</div>
<div data-cy="aut-panel" class="flex-grow h-full bg-gray-100 relative" style="width: 1901px;">
  <div class="p-0 m-0 bg-white" data-v-1af2ca55>...</div>
  <div class="h-0 p-16px" data-v-1af2ca55>
    <!-->
    <div id="unified-runner" class="origin-top viewport" data-v-1af2ca55 style="width: 1000px; height: 660px; transform: scale(0.55303); position: absolute; margin-left: 464.5px;">
      <div class="screenshot-height-container"> == $0
        <iframe id="Your project: 'Test Project'" title="Your project: 'Test Project'" class="aut-iframe" src="http://localhost:4200/">...</iframe>
        <iframe id="Your Spec: '/__cypress/iframes/src%2Fe2e%2Finit.cy.ts'" class="spec-iframe" src="/__cypress/iframes/src%2Fe2e%2Finit.cy.ts">...</iframe>
      </div>
    </div>
    <!-->
    <div class="screenshot-helper-pixels" data-v-0efcdee6 data-v-1af2ca55>...</div>
  </div>
</div>
</div>
```

2 iFrames, same origin

```
cy.visit()
```

```
cy.get()
```

```
cy.click()
```

synchronous



- visit
- get
- click

asynchronous



ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

Commands

- Methods of "cy." object
- Run asynchronously
 - "Feels" like a Promise (thenable) but isn't
 - Don't use async/await
- Are chainable



Common Commands

Queries

- get
- contains
- find
- sibling
- parent

Actions

- click
- type
- clear

Assertions

- should
 - explicit
 - implicit
- and



Implicit Assertions

- Behaves like a normal command
- Does waiting as well
- Good for single assertions
- Chaining with and

```
cy
  .get('h1')
  .should(
    'have.text',
    ' Unforgettable Holidays '
  );
```



Explicit Assertions

- More verbose
- Good for multiple assertions
- Messaging

```
cy.get('h1').should(($h1) => {  
  
  expect($h1).to.have  
  
    .text(' Unforgettable Holidays ');  
  
});
```



cypress.config.ts

```
import { defineConfig } from 'cypress';

export default defineConfig({  
  e2e: {  
    chromeWebSecurity: true,  
    experimentalWebKitSupport: true,  
    retries: { ← Retries (main for backend systems)  
      runMode: 1  
    }  
  },  
  // further configuration properties  
});
```



Lab Time



Selection Strategies



Selector Strategies

DOM-Based

- id Attribute
 - Must be unique
 - Browser-internal meaning
 - Consensus not to use it
- CSS Classes
 - Should be used Layout
 - Don't deal with functionality
- Text
 - cy.contains
 - Better to go with User-Facing Selectors
- data-testid
 - Best Practices
 - Recommended by Cypress

User-Facing

- Use text and/or roles
- Includes partially a11y
- Testing Library Bindings for Cypress
- Careful when text changes often
 - Fallback to data-testid



Testing-Library Selectors

1. **cy.findByRole** ← Buttons, Links, etc. Role is automatically assigned by browser
2. **cy.findByLabelText** ← Form Fields
3. cy.findByPlaceholderText
4. cy.findByText
5. cy.findByDisplayValue
6. cy.findByAltText
7. cy.findByTitle
8. cy.findByTestId




Examples

- `cy.findByRole('link', { name: 'Holidays' })`
`.click();`
- `cy.findByRole('button', { name: 'Save' })`
`.click();`
- `cy.findByLabelText(/Firenze/i);`
- `cy.findByLabelText(/Latitia/)`
`.findByRole('link', { name: 'Edit Customer' });`



Accessibility Extension in Chrome


 **Unforgettable Holidays**

Holidays

Customers

Newsletter

Vienna / Wien
Dive into the capital of the Habsburg empire




ON SALE


With a population of almost 2 million, Vienna is the second largest German-speaking city and breathes history in

Get a Brochure


Detroit
Top 10 of the cosmopolitan cities of the world



Reykjavik
The northernmost capital of the world and entry to Iceland



London
Explore one of the world's most famous cities



Being a tourist magnet, this city needs no further introduction.

Get a Brochure

mat-card.mat-mdc-card.mdc-card.my-4.mx-4.ng-star-inserted 305 x 416

Sign In

Reload required before the change takes effect. Reload DevTools

Elements Console Sources >> 1

RootWebArea "Eternal" focusable: true focused: true

- link "Eternal Unforgettable Holidays" focusable: true
 - heading "Eternal Unforgettable Holidays"
- button "Sign In" focusable: true
 - StaticText "Sign In"
- list ""
 - listitem ""
 - listitem ""
 - listitem ""
- generic "Vienna / Wien"
- generic "London"
- generic "Detroit"
- generic "Reykjavík"
- generic "Copenhagen / København"
 - StaticText "Copenhagen / København"
 - StaticText "Europe's most livable city"
 - figure ""
 - StaticText "Copenhagen is the capital of Denmark and is located on Zealand which is one of the three main Danish islands"

Styles Computed Layout Event Listeners Accessibility >>

Accessibility Tree



ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

Demo



Cypress Goodies

- lodash: Cypress._
- jQuery: Cypress.\$
- Uses Mocha and Chai
 - Not Jest
 - Very similar commands



Why not Cypress?

- ~~Cross Browser Support~~ ← Chromium, Firefox, Webkit (exp.)
- ~~Multiple Super Domains~~ ← Available since v9.6
- Parallelization
- Special Requirements
 - Multiple Tabs
 - WebComponents
 - ...



Flakiness / Element not found

cy

```
.get('a') // Query  
.should('have.attr', 'href') // Assertion  
.contains('Customers') // Query  
.click(); // Action
```

```
1 visit  
2 get a  
3 -assert expected <a.home> to have attribute href  
4 -contains Customers
```

! CypressError

Timed out retrying after 4000ms: cy.contains() failed because it requires a DOM element, window or document.

The subject received was:

> /

The previous command that ran was:

> cy.should()

src/e2e/init.cy.ts:3:43

```
1 | it('should open the browser', () => {  
2 |   cy.visit('');  
> 3 |   cy.get('a').should('have.attr', 'href').contains('Customer  
   |                                     ^  
4 |   // cy.get('button[role=switch]').first().click();  
5 |   cy.get('[data-testid=btn-click]') // Query  
6 |   .click() // Action
```

> View stack trace

Print to console



ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

Flakiness / Undetached DOM Event

cy

```
.get('[data-testid=btn-click])' // Query

.click() // Action

.should('have.text', 'Unclick me'); // Assertion
```

- Assertion after Action
- Not retried
- Might work
 - Depends on the implementation

```
1 visit
2 get [data-testid=btn-click]
3 -click
```

! CypressError

Timed out retrying after 4050ms: cy.should() failed because the page updated as a result of this command, but you tried to continue the command chain. The subject is no longer attached to the DOM, and Cypress cannot requery the page after commands such as cy.should().

Common situations why this happens:

- Your JS framework re-rendered asynchronously
- Your app code reacted to an event firing and removed the element

You can typically solve this by breaking up a chain. For example, rewrite:

```
> cy.get('button').click().should('have.class', 'active')
```

to

```
> cy.get('button').as('btn').click()
> cy.get('@btn').should('have.class', 'active') Learn more
```

[src/e2e/init.cy.ts:7:6](#)

```
5 |   cy.get('[data-testid=btn-click]) // Query
6 |     .click() // Action
> 7 |     .should('have.text', 'Unclick me'); // Assertion
    |     ^
8 |   });
9 |
```

> [View stack trace](#)

[Print to console](#)

Flakiness / Unreachability

cy

```
.get('button[role=switch]')
```

```
.first()
```

```
.click();
```

- Use parent element
 - Doesn't have to be an input/button
- Typical Error in UI Libraries

! CypressError

Timed out retrying after 4050ms: cy.click() failed because this element:

```
<button role="switch" type="button" class="mdc-switch mdc-switch--selected mdc-switch--checked" tabindex="0" id="mat-mdc-slide-toggle-1-button" aria-labelledby="mat-mdc-slide-toggle-1-label" aria-checked="true">...</button>
```

is being covered by another element:

```
<div class="mdc-switch__ripple">...</div>
```

Fix this problem, or use {force: true} to disable error checking. [Learn more](#)

src/e2e/init.cy.ts:4:41

```
2 | cy.visit('');
3 | // cy.get('a').should('have.attr', 'href').contains('Cus
> 4 | cy.get('button[role=switch]').first().click();
   |                                     ^
5 | cy.get('[data-testid=btn-click]') // Query
6 | .click() // Action
7 | .should('have.text', 'Unclick me'); // Assertion
```

> View stack trace

Print to console