

DotNet 2021

ONLINE TECH CONFERENCE

22nd June 2021

#DotNet2021

Perdiendo el miedo a las pruebas e2e con Cypress

www.dotnet2021.com



DotNet 2021

ORGANIZATION

plain
concepts

IN COOPERATION WITH

FUNDACIÓN
GOMAESPUMA
"Educando con una sonrisa."

SPONSORS

 Microsoft

DevsDNA™ 

intelequia

 My Public[®]
Inbox

#DotNet2021



Asun Sánchez

Software Engineer

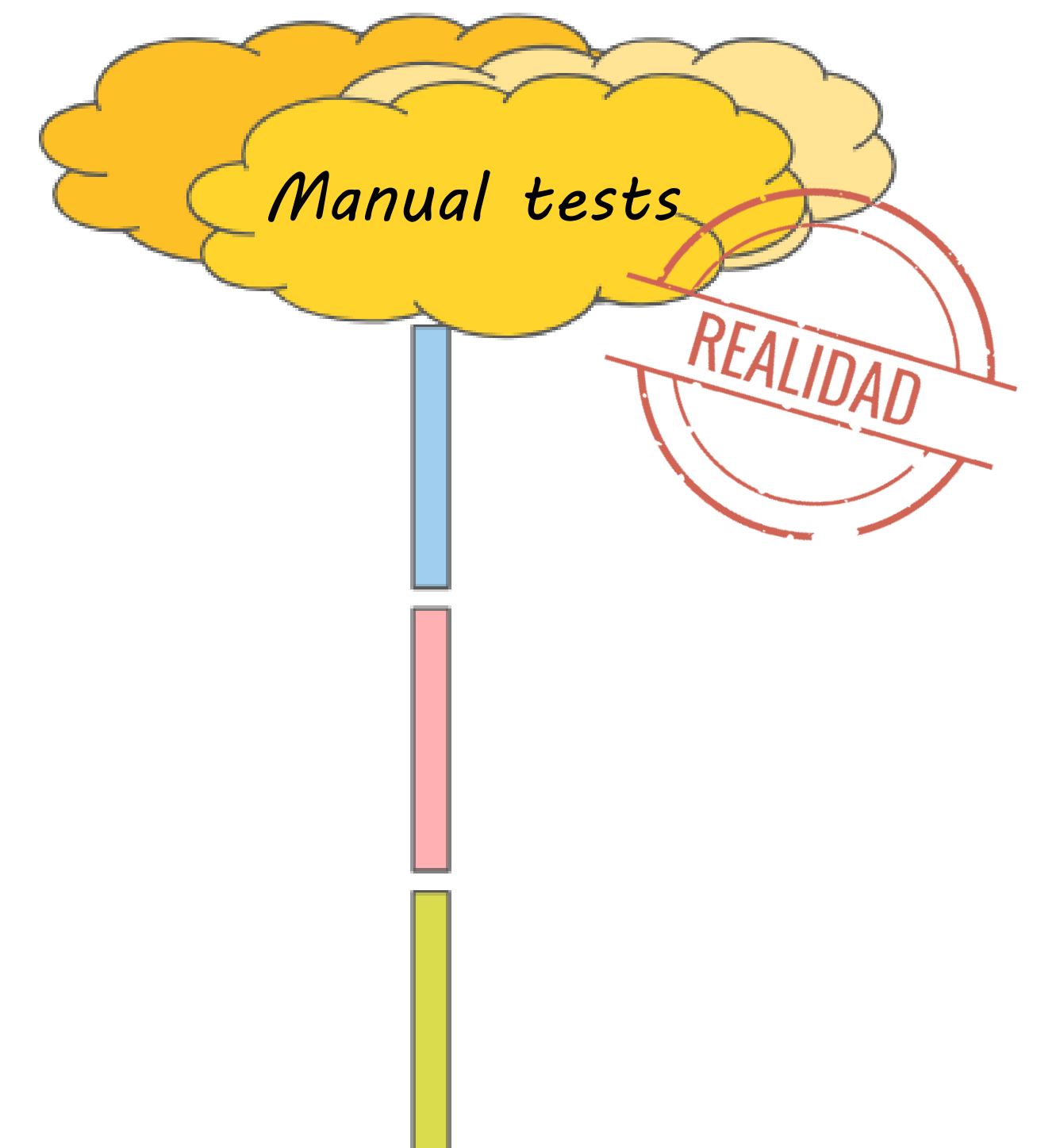
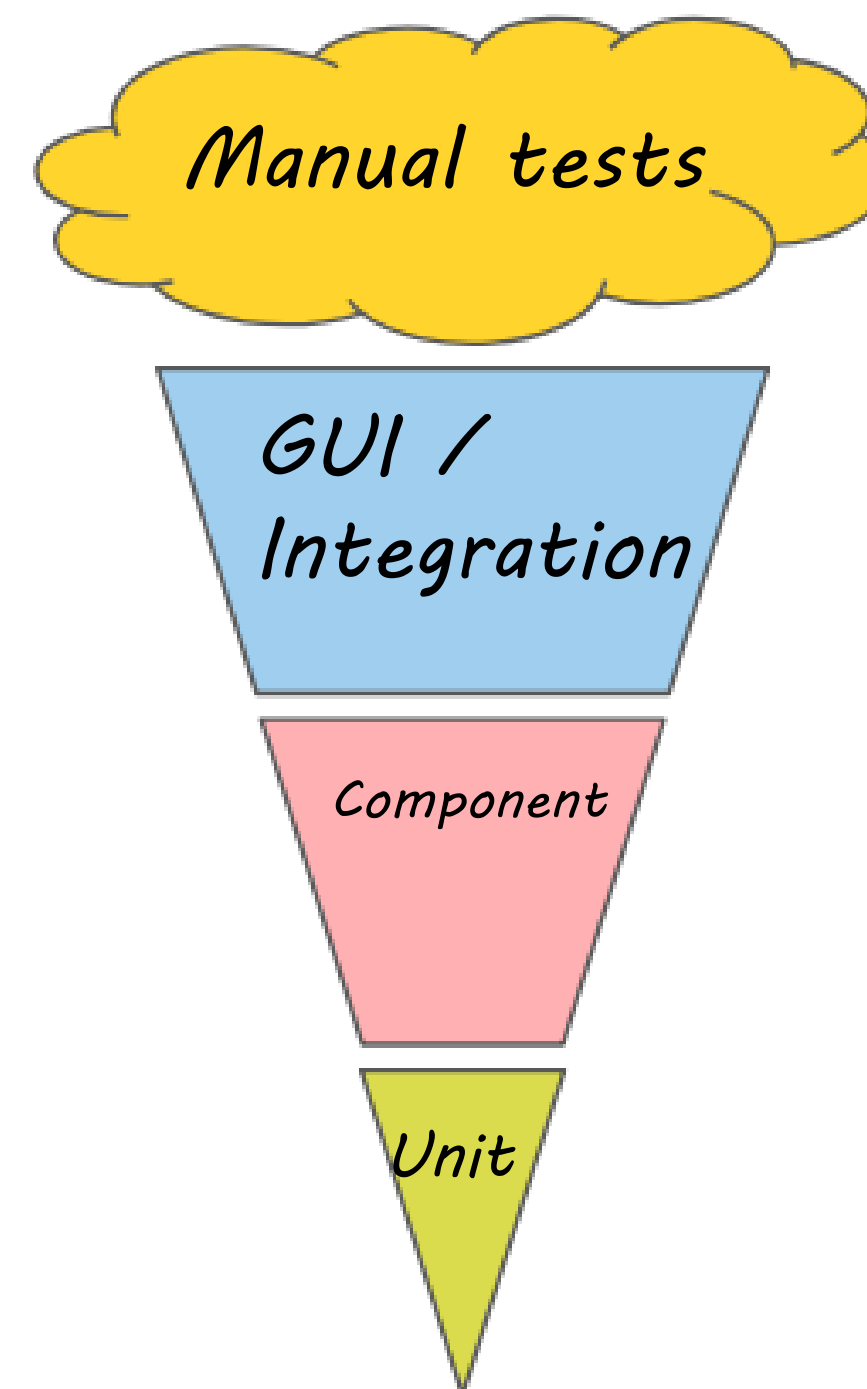
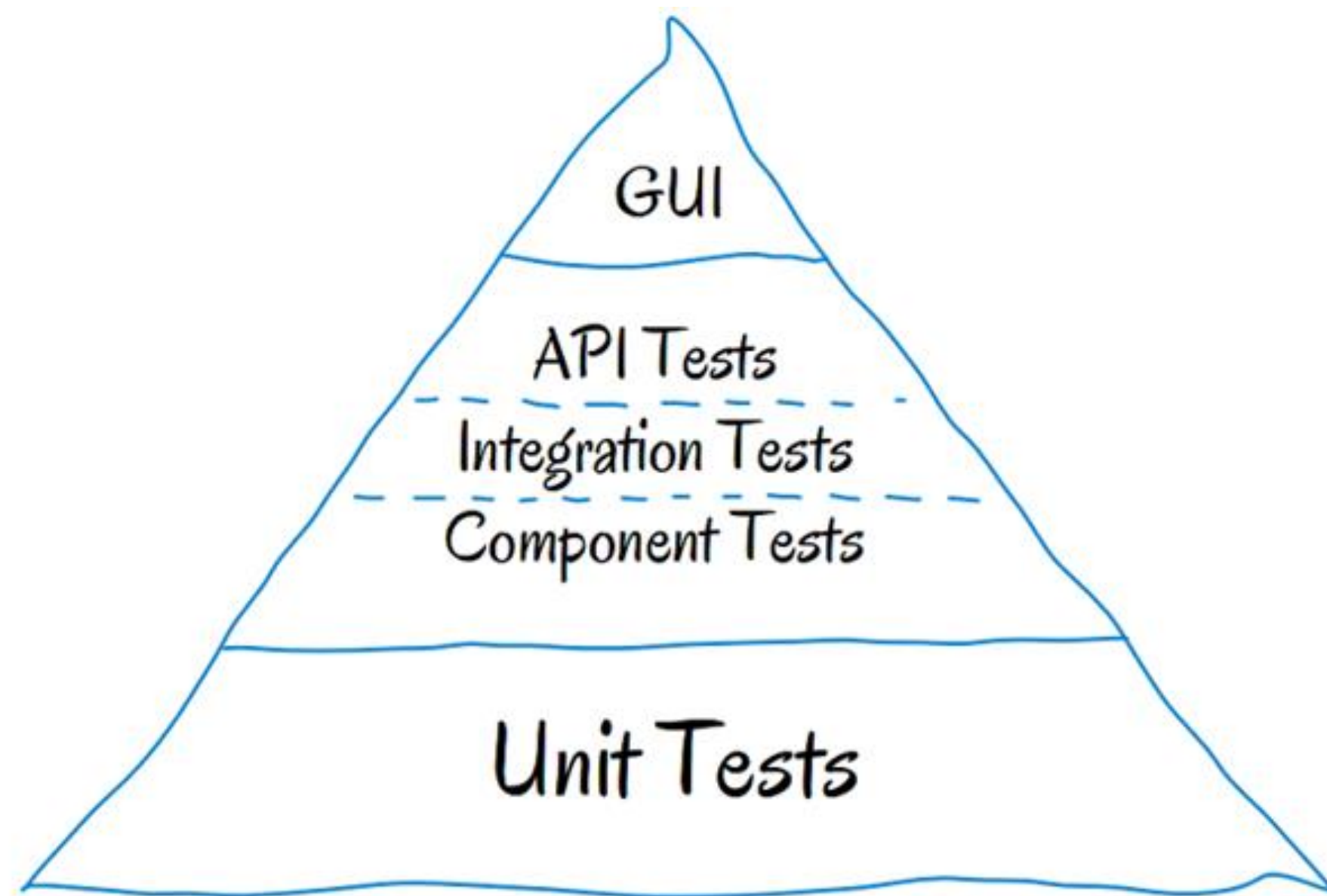
@asunsanlop
aslopez@plainconcepts.com



Iván Reinoso

Frontend Technical Lead

@ivanirega
ireinoso@plainconcepts.com



THE FOUR TYPES OF TESTS

End to End

A helper robot that behaves like a user to click around the app and verify that it functions correctly.

Sometimes called “functional testing” or e2e.

Integration

Verify that several units work together in harmony.

Unit

Verify that individual, isolated parts work as expected.

Static

Catch typos and type errors as you write the code.





“Fast, easy and reliable testing for anything that runs in a browser.”

Front-End, browser testing

Todo tipo de tests, enfocado en e2e testing

IDE entorno para ejecución y debug

Integración con CI/CD con headless

Herramienta Framework agnostic

The screenshot shows the Cypress.io website with the following content:

- Header:** Cypress.io logo, navigation links (Commands, Utilities, Cypress API), and a search bar.
- Section: Actions**
 - Examples of actions being performed on DOM
 - docs.cypress.io
- .type()**
 - To type into a DOM element, use the `.type()` command.
 - Code examples:

```
cy.get('.action-email')
  .type('fake@email.com').should('have.value', 'fake@email.com')

// .type() with special character sequences
.type('{leftarrow}{rightarrow}{uparrow}{downarrow}')
.type('{del}{selectall}{backspace}')
```
 - Code examples:

```
// .type() with key modifiers
.type('{alt}{option}') //these are equivalent
.type('{ctrl}{control}') //these are equivalent
.type('{meta}{command}{cmd}') //these are equivalent
.type('{shift}')
```
 - Code examples:

```
// Delay each keypress by 0.1 sec
.type('slow.typing@email.com', { delay: 100 })
.should('have.value', 'slow.typing@email.com')
```
- Actions List:**
 - ✓ `.type()` - type into a DOM element
 - ✓ `.focus()` - focus on a DOM element
 - BEFORE EACH
 - 1 visit https://example.cypress.io/commands/actions
 - TEST BODY
 - 1 get .action-focus
 - 2 - focus
 - 3 - assert expected <input#password1.form-control.action-focus.focus> to have class focus
 - 4 - prev
 - 5 - assert expected <label> to have attribute style with the value color: orange;
 - ✓ `.blur()` - blur off a DOM element
 - ✓ `.clear()` - clears an input or textarea element
 - ✓ `.submit()` - submit a form
 - ✓ `.click()` - click on a DOM element
 - ✓ `.dblclick()` - double click on a DOM element
 - ✓ `.rightclick()` - right click on a DOM element
 - ✓ `.check()` - check a checkbox or radio element
 - ✓ `.unchecked()` - uncheck a checkbox element
 - ✓ `.select()` - select an option in a <select> element
 - ✓ `.scrollIntoView()` - scroll an element into view
 - ✓ `.trigger()` - trigger an event on a DOM element
 - ✓ `cy.scrollTo()` - scroll the window or element to a position

Instalación y configuración



```
npm install cypress --save-dev
```

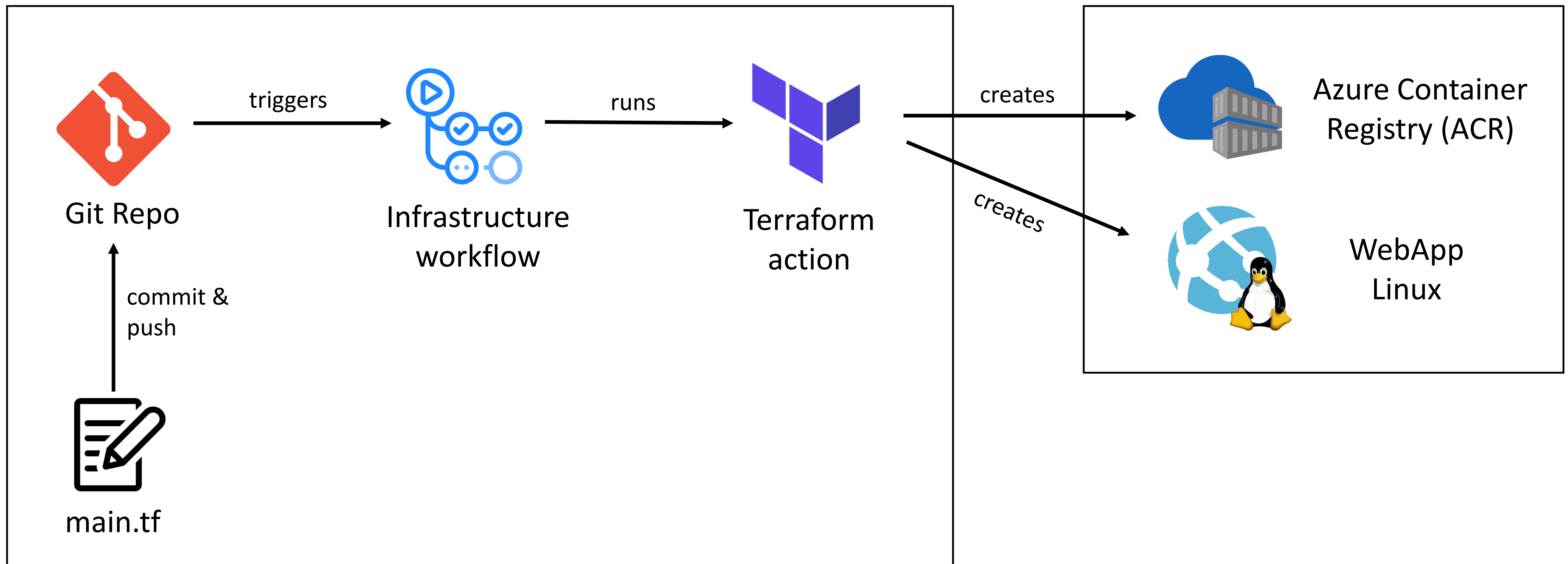
```
npx cypress open
```

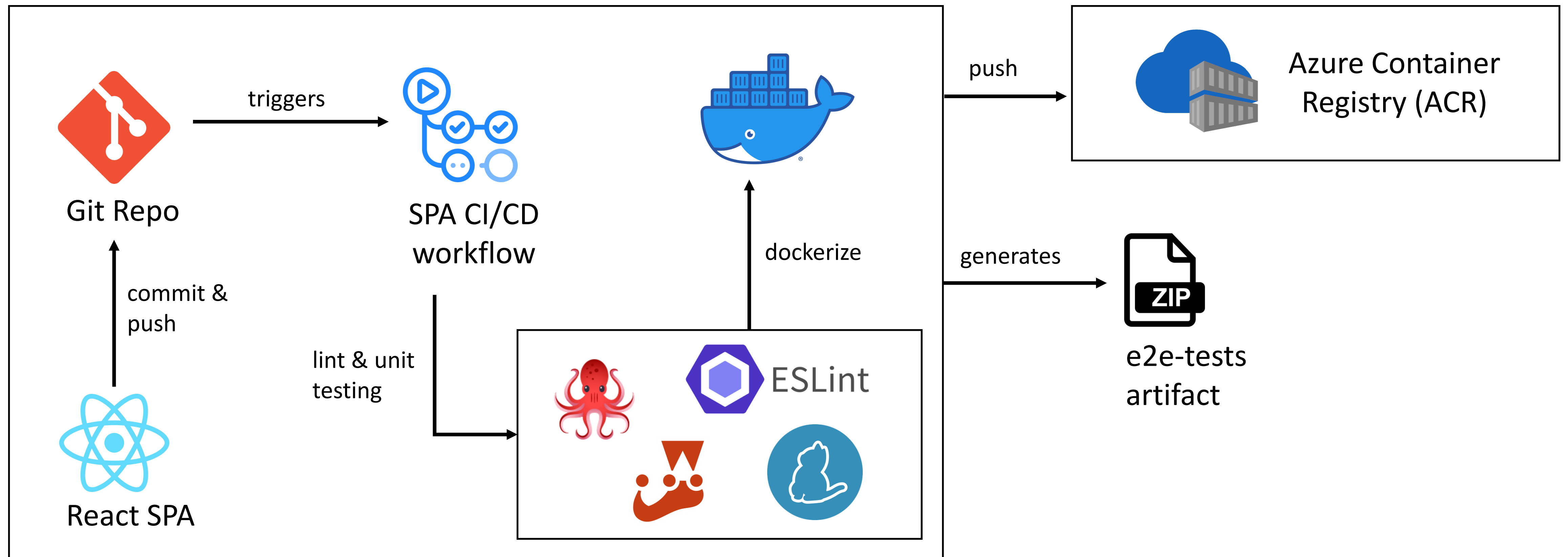
```
yarn add cypress --dev
```

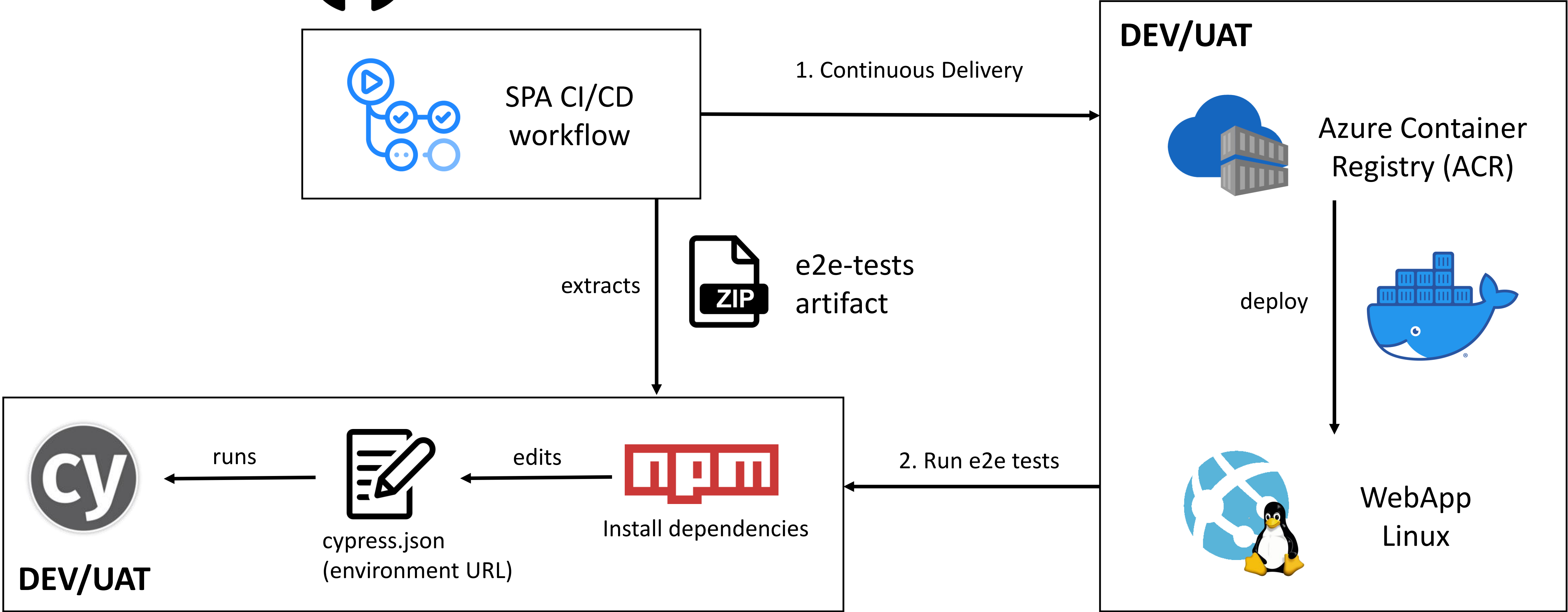
```
yarn run cypress open
```

+ typescript? .tsconfig.json:

```
{
  "compilerOptions": {
    "target": "es5",
    "lib": [
      "es5",
      "dom"
    ],
    "types": [
      "cypress"
    ]
  },
  "include": [
    "**/*.ts"
  ]
}
```







Recapitulando...



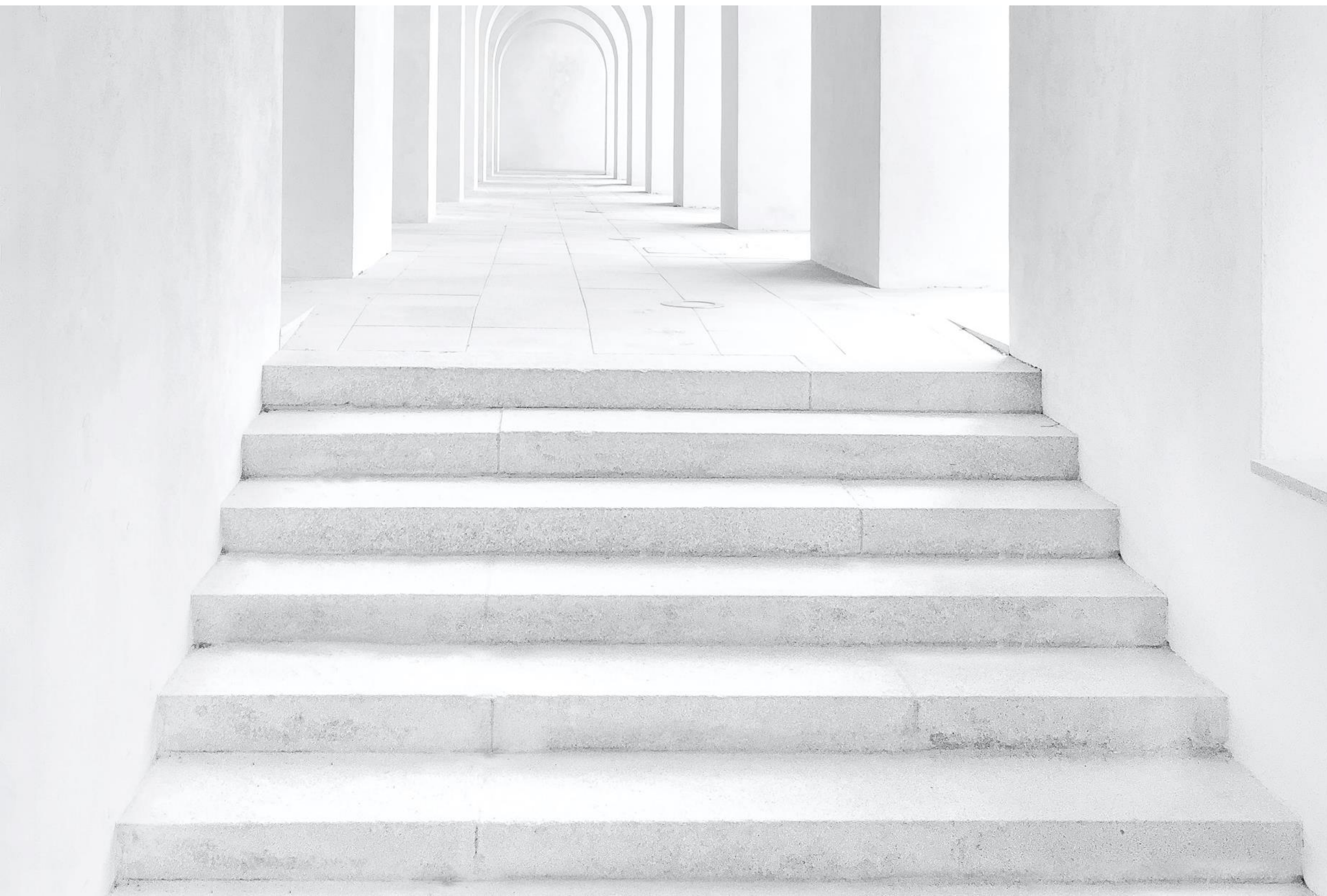
Recordamos **por qué** hacer testing

Utilizamos sólo dos comandos para **instalación y configuración** de cypress

Empezamos con tests e2e: **visitar**, recuperar elementos página, **interactuar**

Integración continua con nuestros tests

Siguientes pasos



Doc. oficial: <https://www.cypress.io/>

Incluir **component testing**

Cubrir **mayor % funcionalidad**

Siempre probar **UAT o DEV**, nunca PRO

Investigar aspectos avanzados: login (ROPC),
variables de entorno, commands, testing strategies...

DotNet 2021

ONLINE TECH CONFERENCE

www.dotnet2021.com

#DotNet2021

Thanks and ... See you soon!

Thanks also to the sponsors. Without whom this would not have been posible.

plain
concepts

FUNDACIÓN
GOMAESPUMA
"Educando con una sonrisa."

Microsoft

intelequia



My Public
Inbox

