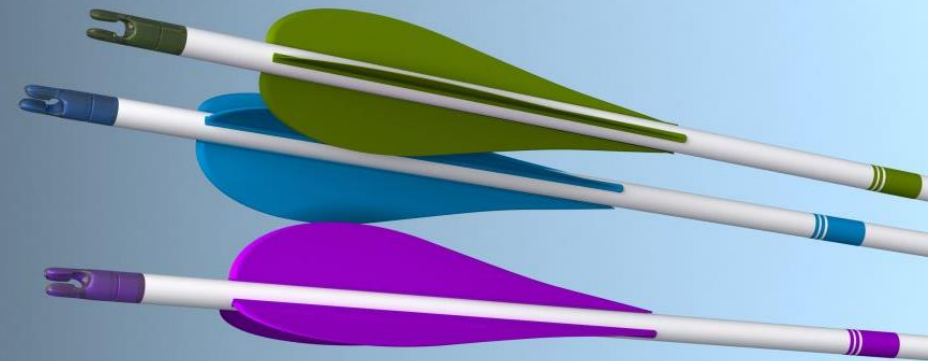


Testowanie Automacyjne II

WYKŁAD 1

O zaliczeniu

- ▶ Warunkiem zaliczenia jest dostarczenie projektu testów automatycznych napisanych przy użyciu narzędzia Cypress
- ▶ Należy dostarczyć do 23.06.2023
 - ▶ Opis testów
 - ▶ Wynik testów
 - ▶ Repozytorium z testami
 - ▶ Jeżeli testy opierają się na waszym projekcie to wymagane elementy do jego uruchomienia



O zaliczeniu bazowa ocena

► Ocena

- 3.0 Należy dostarczyć 10 testów automatycznych
 - Każdy test musi się składać z akcji i asercji
 - Testy muszą przechodzić



O zaliczeniu poprawienie oceny

- ▶ Dodatkowe +0,5 do oceny za każde z:
 - ▶ Rozszerzenie raportowania testów przez raport HTML
 - ▶ Fixtures
 - ▶ Użycie cypress.json do konfiguracji
 - ▶ Tagi
 - ▶ PageObject
 - ▶ Praca w zespole
 - ▶ wzrost ilości testów (12 testów per osoba)
 - ▶ Dashboard

Czym jest Cypress?

- ▶ Jest to narzędzie testów Automatycznych używające języka JavaScript oraz Node.js
- ▶ Składa się z:
 - ▶ Cypress Test Runner
 - ▶ Cypress Dashboard



Cypress

- ▶ Jak działa Cypress?
 - ▶ W maszynie wirtualnej node.js jest wystawiony uruchomiony test runner.
 - ▶ Test runner jest odpolany w postaci aplikacji web.
 - ▶ Gdy uruchomimy grupę testów zostanie uruchomiony w nowym oknie (z linii poleceń w trybie headless)
 - ▶ Oznacza to że egzekucja kodu jest od wewnątrz przeglądarki



Trochę o wersjach Cypress

- ▶ Wersja 10+ (najnowsza wersja to 12.10)
 - ▶ Zmieniła:
 - ▶ Konfigurowanie
 - ▶ Natywne działanie i wybór plików testowych
 - ▶ Kompatybilność z paczkami
 - ▶ Usunęła plugin file i przeniósł jego funkcjonalność do nowego config
- ▶ Wersja 9.7
- ▶ Co robić? Jak żyć?

Choose a testing type — [Need help](#) ?



E2E Testing

Build and test the entire experience of your application from end-to-end to ensure each flow matches your expectations.

● Running

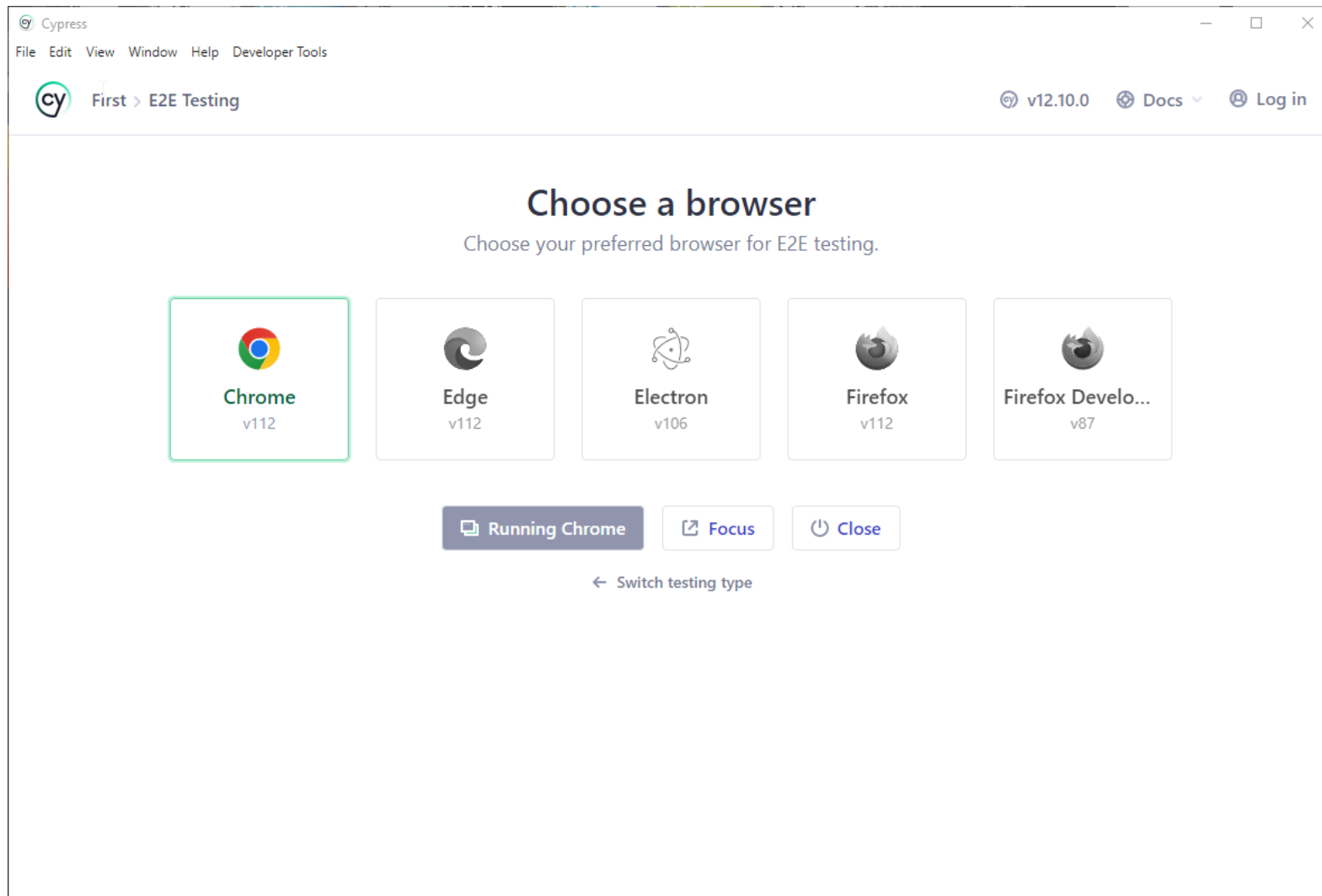


Component Testing

Build and test your components from your design system in isolation in order to ensure each state matches your expectations.

○ Not Configured

Cypress UI



Cypress UI

First x +

localhost:52452/_/#/specs

Chrome is being controlled by automated test software.

First

Specs

Runs

Debug

Settings

Specs

Search specs

20 matches

+ New spec

E2E specs

Last updated ?

Latest runs ?

Average duration ?

▼ cypress\e2e

▼ 1-getting-started

todo.cy.js a few secs ago -- --

▼ 2-advanced-examples

actions.cy.js a few secs ago -- --

aliasing.cy.js a few secs ago -- --

assertions.cy.js a few secs ago -- --

connectors.cy.js a few secs ago -- --

cookies.cy.js a few secs ago -- --

cypress_api.cy.js a few secs ago -- --

files.cy.js a few secs ago -- --

location.cy.js a few secs ago -- --

misc.cy.js a few secs ago -- --

navigation.cy.js a few secs ago -- --

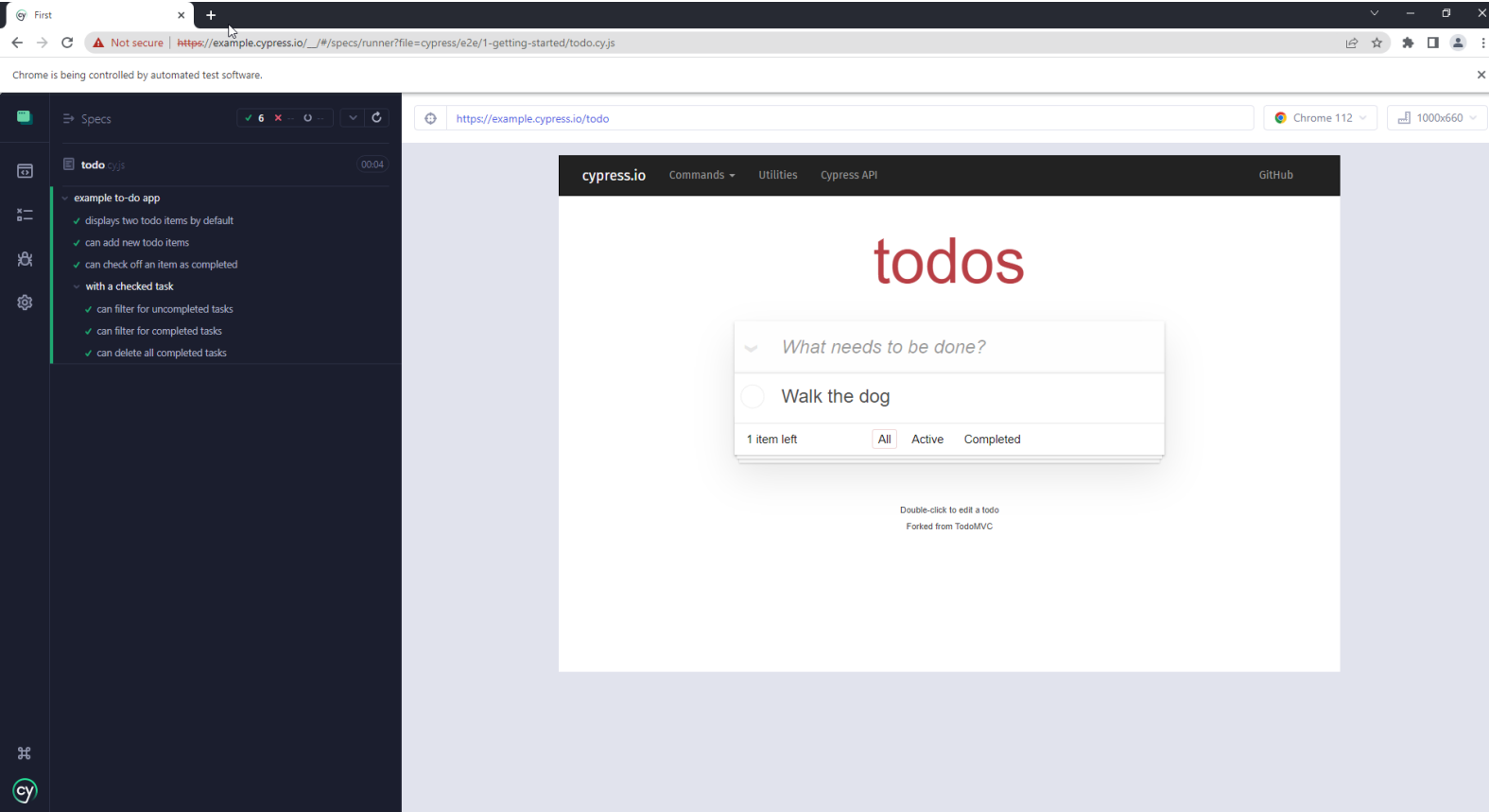
network_requests.cy.js a few secs ago -- --

querying.cy.js a few secs ago -- --

spies_stubs_clocks.cy.js a few secs ago -- --

cy

Cypress UI



Cypress UI

Cypress Przeznaczenie

- ▶ Testów End-to-End
- ▶ Testów Integracyjnych
- ▶ Testów Jednostkowych



Ałuty



- ▶ Co daje nam cypress:
 - ▶ Szybkie, skuteczne i wygodne narzędzie
 - ▶ Cypress umożliwia łatwe zrównoleglenie testów
 - ▶ Wbudowane cofanie się w krokach testów
 - ▶ Wbudowane nagrywanie wyników przejścia testów
 - ▶ Obniżona zależność od wersji przeglądarki
 - ▶ Brak zależności Webdriver – Browser
 - ▶ Rozszerzalność

Wady



- ▶ Co cypress nam uniemożliwia:
 - ▶ Działania na wielu zakładkach
 - ▶ Działania na wielu przeglądarkach
 - ▶ Działania na cross-origin
 - ▶ Eksperymentalnie od wersji 12

Wymagania



- ▶ Co jest wymagane do działania cypress:
- ▶ System operacyjny
 - ▶ macOS 10.9+(64bit)
 - ▶ Ubuntu 12.04+(64bit)
 - ▶ Windows 7+(64bit)
- ▶ Node.js
 - ▶ Wraz npm
- ▶ Edytor wspierający składnię JS
 - ▶ Visual Studio Code

Instalacja npm



- ▶ Polecenie instalacyjne node js:
 - ▶ Bazowe:
 - ▶ `npm install cypress`
 - ▶ Dołączanie do istniejącego jako element developerski
 - ▶ `npm install cypress --save-dev`
- ▶ Pierwsze uruchomienie
 - ▶ `./node_modules/.bin/cypress open`
 - ▶ `npx cypress open`

Często spotykane problemy



- ▶ Najczęstszy problem w Windows
 - ▶ Brak możliwości wykonywania skryptów
 - ▶ Umożliwienie Egzekucji skryptów:
 - ▶ Set-ExecutionPolicy RemoteSigned
 - ▶ Polecenie "npx cypress open" nie wymaga uprawnienia do uruchamiania skryptów

UI

File Edit View Window Help Developer Tools



First > E2E Testing ¹

²

v12.10.0

³

Docs

⁴

Log in

Choose a browser

⁵

Choose your preferred browser for E2E testing.



Chrome
v112



Edge
v112



Electron
v106



Firefox
v112



Firefox Develo...
v87

⁶

Running Chrome

Focus

Close

← Switch testing type

UI

First x +

Not secure | https://example.cypress.io/_/#/specs

Chrome is being controlled by automated test software.

7 v12.10.0 Chrome 112 Docs Log in

1 First

2 Specs

3 Runs

4 Debug

5 Settings

Specs

Search specs 20 matches + New spec

6 E2E specs

Last updated ? Latest runs ? Average duration ?

cyypress\e2e

1-getting-started

todo.cyjs 12 mins ago -- --

2-advanced-examples

actions.cyjs 12 mins ago -- --

aliasing.cyjs 12 mins ago -- --

assertions.cyjs 12 mins ago -- --

connectors.cyjs 12 mins ago -- --

cookies.cyjs 12 mins ago -- --

cypress_api.cyjs 12 mins ago -- --

files.cyjs 12 mins ago -- --

location.cyjs 12 mins ago -- --

misc.cyjs 12 mins ago -- --

navigation.cyjs 12 mins ago -- --

network_requests.cyjs 12 mins ago -- --

querying.cyjs 12 mins ago -- --

spies_stubs_clocks.cyjs 12 mins ago -- --

location.cyjs

https://example.cypress.io/_/#/specs/runner?file=cypress/e2e/2-advanced-examples/location.cyjs

UI

The image shows the Cypress test runner interface. On the left, the 'Specs' panel displays a test suite named 'todo cy.js' with a duration of 00:02. The suite contains two sections: 'example to-do app' and 'with a checked task'. The 'example to-do app' section has three tests: 'displays two todo items by default', 'can add new todo items', and 'can check off an item as completed'. The 'with a checked task' section has three tests: 'can filter for uncompleted tasks', 'can filter for completed tasks', and 'can delete all completed tasks'. All tests are marked as passed with green checkmarks. The main panel shows the application under test, which is a todo app running in a Chrome browser. The app has a header with 'cypress.io', 'Commands', 'Utilities', 'Cypress API', and 'GitHub'. The main content area displays the word 'todos' in a large red font. Below it is a form with a placeholder 'What needs to be done?' and a single todo item 'Walk the dog' with an unchecked checkbox. At the bottom, it shows '1 item left' and filter buttons for 'All', 'Active', and 'Completed'. A footer message says 'Double-click to edit a todo' and 'Forked from TodoMVC'.

Chrome is being controlled by automated test software.

Specs

6

todo cy.js 00:02

- example to-do app
 - displays two todo items by default
 - can add new todo items
 - can check off an item as completed
- with a checked task
 - can filter for uncompleted tasks
 - can filter for completed tasks
 - can delete all completed tasks

https://example.cypress.io/todo

Chrome 112 1000x660

cypress.io Commands Utilities Cypress API GitHub

todos

What needs to be done?

☐ Walk the dog

1 item left All Active Completed

Double-click to edit a todo
Forked from TodoMVC

UI

The screenshot displays the Visual Studio Code interface with the following components:

- EXPLORER:** Shows the file structure of a Cypress project. Red callouts 1 through 8 highlight specific files: 1. `cypress.config.js`, 2. `cypress` folder, 3. `downloads` folder, 4. `e2e` folder, 5. `fixtures` folder, 6. `support` folder, 7. `node_modules` folder, and 8. `package-lock.json`.
- EDITOR:** Displays the content of `cypress.config.js`. The code is as follows:

```
1 const { defineConfig } = require("cypress");
2
3 module.exports = defineConfig({
4   e2e: {
5     setupNodeEvents(on, config) {
6       // implement node event listeners here
7     },
8   },
9 });
10
```
- TERMINAL:** Shows the output of a Cypress command, including various GET requests and their response times, such as:

```
CGARHTsZgIgQgASACKAEiAggB8fct=application/x-protobuf&key=AizaSy80ti4mM-6x9WdnZIjIeyEU210pBXqWBgw 200 595.742 ms - -
GET /_/_/assets/Index-4cca605a.css 200 4.365 ms - -
GET /_/_/assets/paper-airplane_x16-862e8833.js 200 3.738 ms - -
GET /_/_/assets/ResultCounts.vue_vue_type_script_setup_true_lang-340207eb.js 200 3.346 ms - -
GET /_/_/assets/SpecNameDisplay.vue_vue_type_script_setup_true_lang-1a441c26.js 200 3.773 ms - -
GET /_/_/assets/settings_x16-8e959a13.js 200 3.261 ms - -
GET /_/_/assets/Index-6ccb669b.js 200 3.137 ms - -
GET /v1/pages/ChVdAhDvbmUMTEYlJAuNTYxNS4xMjISEAnATr7cC86eOxIFDRNQn9Q=?alt=proto 200 586.272 ms - -
GET /_/_cypress/iframes/cypress%5Ce2e%5C1-getting-started%5Ctodo.cy.js 200 0.531 ms - 984
GET /_/_cypress/tests?p=cypress%5Csupport%5Ce2e.js 200 3.634 ms - -
GET /_/_cypress/tests?p=cypress%5Ce2e%5C1-getting-started%5Ctodo.cy.js 200 2.384 ms - -
GET /todo 200 2.571 ms - -
GET /todo 200 2.253 ms - -
GET /todo 200 2.697 ms - -
GET /todo 200 2.438 ms - -
GET /todo 200 2.383 ms - -
GET /todo 200 2.377 ms - -
```
- STATUS BAR:** Indicates the current file is `Ln 1, Col 1`, using `Spaces: 2`, `UTF-8` encoding, `LF` line endings, and `JavaScript` language.

Stand-alone



- ▶ Alternatywna Opcja
 - ▶ Cypress pobieranie bezpośrednie
 - ▶ Ma to służyć do szybkiego sprawdzenia narzędzia a nie do pełnego developmentu
 - ▶ Nie ma opcji nagrywania testów do Dashboard

Składnia

- ▶ Język opisowy testów z którego korzysta cypress to mocha.js
 - ▶ Zatem używamy tzw. Hooks do opisywania
 - ▶ Do dyspozycji mamy:
 - ▶ Describe
 - ▶ Context
 - ▶ It
 - ▶ Before
 - ▶ BeforeEach
 - ▶ After
 - ▶ AfterEach



Przykład 1

cypress > integration > e2e > JS basics.spec.js > ...

```
1  <reference types="cypress" />
2
3  describe('Sample test', function(){
4    it('This is true body of Sample test', function(){
5      expect(true).to.equal(true);
6      cy.visit("http://www.google.com");
7    })
8  })
9
10 describe('Sample test2', () => {
11   it('This is true body of Sample test2', ()=>{
12     expect(true).to.equal(true);
13     cy.visit("http://www.google.com");
14   })
15 })
16 |
```

I

Przykład 2

cypress > integration > e2e > JS basics.spec.js > describe('Sample test2 description') callback > context('Sample test2 context') callback > it('This is true body of S

```
1  /// <reference types="cypress" />
2  context('Sample test Context', function () {
3    describe('Sample test Description', function () {
4      it('This is true body of Sample test', function () {
5        expect(true).to.equal(true);
6        cy.visit("http://www.google.com");
7      })
8    })
9  })
10 describe('Sample test2 description', () => {
11   context('Sample test2 context', () => {
12     it('This is true body of Sample test2', () => {
13       expect(true).to.equal(true);
14       cy.visit("http://www.google.com");
15     })
16   })
17 })
18
19 describe('Sample test3 description', () => {
20   describe('Sample test3 description2', () => {
21     it('This is true body of Sample test3', () => {
22       expect(true).to.equal(true);
23       cy.visit("http://www.google.com");
24     })
25   })
26 })
```

Przykład Działania 1

Tests

✓ 3

✗ --

○ --

05.43

● ↑ ↺

🔒

https://www.google.com/?gws_rd=ssl

1000 x 660 (100%)

cypress/integration/e2e/basics.spec.js

▼ Sample test Context

▼ Sample test Description

✓ This is true body of Sample test

▼ Sample test2 description


▼ Sample test2 context

✓ This is true body of Sample test2

▼ Sample test3 description

▼ Sample test3 description2

✓ This is true body of Sample test3



PL

Zaloguj się

Zanim przejdziesz do wyszukiwarki Google

Google wykorzystuje [pliki cookie](#) i inne dane do:

- ✖ Dostarczania i utrzymywania usług, na przykład do śledzenia przerw w ich działaniu lub do ochrony użytkowników przed spamem, oszustwami i nadużyciami
- 📊 Pomiaru zaangażowania odbiorców i generowania statystyk na temat witryn w celu lepszego zrozumienia, jak są używane usługi Google

Jeśli się na to zgodzisz, pliki cookie i inne dane będziemy wykorzystywać także do:

- 🔍 poprawiania jakości obecnych usług i opracowywania nowych;
- 📊 Wyświetlania reklam i mierzenia ich skuteczności
- ✦ Wyświetlania spersonalizowanych treści (w zależności od Twoich ustawień)
- 📄 Wyświetlania spersonalizowanych lub niespersonalizowanych reklam (w zależności od Twoich ustawień) w usługach Google i w innych miejscach w internecie

Reklamy i treści niespersonalizowane mogą być dobierane na podstawie czynników takich jak aktualnie przeglądane treści czy Twoja lokalizacja (wyświetlanie reklam jest oparte na przybliżonej lokalizacji). Reklamy i treści spersonalizowane również są dobierane według tych czynników oraz dodatkowo na podstawie Twojej aktywności, na przykład haseł wyszukiwanych w Google czy filmów oglądanych w YouTube. Reklamy i treści spersonalizowane to między innymi trafniejsze wyniki i rekomendacje, indywidualna strona główna w YouTube oraz reklamy dopasowane do Twoich zainteresowań.

Jeśli chcesz przejrzeć dostępne możliwości, w tym opcje rezygnacji z plików cookie używanych do personalizacji lub informacje o opcjach na poziomie przeglądarki pozwalających zrezygnować z niektórych lub wszystkich plików cookie używanych w innych celach, kliknij „Dostosuj”. W każdej chwili możesz też wejść na stronę

Assercje

- ▶ Język asercji z którego korzysta cypress to chai.js
 - ▶ Do dyspozycji mamy:
 - ▶ Asercje typu BDD
 - ▶ Expect
 - ▶ Wolna funkcja
 - ▶ Should
 - ▶ Łańcuhowa funkcja
 - ▶ Asercje typu TDD
 - ▶ Assert



Przykład 3

cypress > integration > e2e > JS basics2.spec.js > ...

```
1  /// <reference types="cypress" />
2
3  describe('Sample test using expect', function () {
4    it('Test', function () {
5      expect(true).to.equal(true);
6    })
7  })
8
9  describe('Sample test using should', () => {
10    it('Test', () => {
11      cy.visit("https://pl.wikipedia.org/");
12      cy.url().should('include', "https://pl.wikipedia.org/");
13    })
14  })
15
16
17  describe('Sample test using assert', () => {
18    it('Test', () => {
19      assert.equal(true, true, 'true is true');
20    })
21  })
22
```

I

Przykład Działania 2

Tests ✓ 3 ✗ -- ⌂ -- 00.75

cypress/integration/e2e/basics2.spec.js

Sample test using expect

✓ Test

TEST BODY

```
1 assert expected true to equal true
```

Sample test using should

✓ Test

TEST BODY

```
1 visit https://pl.wikipedia.org/ -> 301: https://pl.wikipedia.or...
2 url
3 - assert expected https://pl.wikipedia.org/wiki/Wikipedia:Strona_g%C5%82%C3%B3wna to include https://pl.wikipedia.org/
```

Sample test using assert

✓ Test

TEST BODY

```
1 assert true is true: expected true to equal true
(xhr) GET 200 /w/api.php?action=query&format=json&formatversion=2&usel...
(xhr) GET 200 /w/api.php?action=discussiontool&pageinfo&format=json&fo...
(xhr) GET 200 /w/api.php?action=visualeditor&format=json&formatversion...
(xhr) GET 200 /w/api.php?action=query&format=json&formatversion=2&usel...
(xhr) GET 200 /w/api.php?action=discussiontool&pageinfo&format=json&fo...
(xhr) GET 200 /w/api.php?action=visualeditor&format=json&formatversion...
(xhr) GET 200 /api/rest_v1/page/summary/Wacław_Kuczajowski
(xhr) GET 200 /api/rest_v1/page/summary/Wac%C5%82aw_Kuczajow...
```

https://pl.wikipedia.org/wiki/Wikipedia:Strona_g%C5%82%C3%B3wna 1000 x 600

WIKIPEDIA
Wolna encyklopedia

Strona główna Dyskusja Czytaj Kod źródłowy Historia i autorzy Przeszukaj Wikipedię

Nie jesteś zalogowany Dyskusja Edycje Utwórz konto Zaloguj się

MIEDZYNARODOWY DZIEŃ MUZEÓW
Pomóż nam rozwijać otwartą wiedzę o kulturze.

KONKURS WIKIDATA OD 4 MAJA DO 18 MAJA 2022
Weź udział i wygrywaj nagrody!

Witaj w **Wikipedii**,
wolnej encyklopedii, którą każdy może redagować.
1 520 747 artykułów, w tym 4479 wyróżnionych

Nauki ścisłe i przyrodnicze • Nauki społeczne i humanistyczne • Technika • Geografia • Historia • Kultura • Społeczeństwo • Sport • Religia • Polska
Więcej portali • Indeks haseł • Przegląd kategorii

Nagrody kaskadowe czekają na uczestników konkursu **CEE Spring 2022!**

Czy wiesz...
Z **nowych** i ostatnio rozbudowanych artykułów w Wikipedii:
...że brytyjski „**ciągnik inżynieryjny**” (na zdjęciu) potrafi pływać?
...w jakim celu powstała i jakie faktycznie cele realizuje **organizacja** kierowana przez

Artykuł na medal
Avengers – amerykański fantastycznonaukowy film akcji z 2012 roku na podstawie serii komiksów o grupie superbohaterów o tej samej nazwie wydawnictwa **Marvel Comics**. Za reżyserię i za scenariusz odpowiadał **Joss Whedon**. W głównych rolach wystąpili

Strona główna Losuj artykuł Kategorie artykułów Najlepsze artykuły Częste pytania (FAQ)
Dla czytelników O Wikipedii Kontakt Wspomóż Wikipedię
Dla wikipedystów Pierwsze kroki Portal wikipedystów Ogłoszenia Zasady Pomoc Ostatnie zmiany
Narzędzia Linkujące Zmiany w linkowanych Prześlij plik Strony specjalne

Polecenia

- ▶ Co do samych poleceń cypress
 - ▶ Mamy do dyspozycji funkcje główne
 - ▶ `cy.visit()`
 - ▶ `visit` ma jedno ograniczenie
 - ▶ Nie można odwiedzać więcej niż jednej domeny w ramach funkcji `it()`
 - ▶ `cy.get()`
 - ▶ Nasze główne narzędzie do wybierania elementów
 - ▶ Działa na selektorach CSS
 - ▶ `cy.url()`
 - ▶ Pobiera obecny url



Polecenia Interakcji

- ▶ Co do samych poleceń cypress
 - ▶ Mamy do dyspozycji funkcje łańcuchowe wykonujące akcje
 - ▶ `.click()`
 - ▶ Stare dobre kliknięcie
 - ▶ `.clear()`
 - ▶ Czyści input wspierający polecenie `clear`
 - ▶ `.type()`
 - ▶ Wpisywanie tekstu obsługuje symulacje przycisków
 - ▶ `.submit()`
 - ▶ Wysyła submit na form
 - ▶ `.focus()`
 - ▶ Ustawia focus na elemencie



Przykład 4

ss > integration > e2e > JS basics3.spec.js > ...

```
/// <reference types="cypress" />
```

```
describe('User can close cookie popup', function () {
  it('Close by ok button', function () {
    cy.visit('http://www.google.com');
    cy.viewport(1920,1080);
    cy.get('#L2AGLb > .QS5gu').should('be.visible');
    cy.get('#L2AGLb > .QS5gu').click();
    cy.get('#L2AGLb > .QS5gu').should('not.be.visible');
  })
  it('Personalize should be available', function () {
    cy.visit('http://www.google.com');
    cy.viewport(1920,1080);
    cy.get('#VnjCcb > .QS5gu').should('be.visible');
    cy.get('#VnjCcb > .QS5gu').click();
    cy.url().should('include', 'https://consent.google.com/');
  })
})
```


Polecenia Szukające

- ▶ Co do samych poleceń cypress
 - ▶ Mamy do dyspozycji funkcje poruszania się po DOM
 - ▶ `.children()`
 - ▶ Wszystkie dzieci danego obiektu
 - ▶ `.parents()`
 - ▶ Wszyscy rodzice danego obiektu
 - ▶ `.next()`
 - ▶ Najbliższy brat/siostra obiektu
 - ▶ `.find()`
 - ▶ Element w strukturze obiektu o specyficznym selektorze
 - ▶ `.filter()`
 - ▶ Od filtrowanie elementów
 - ▶ `.within()`
 - ▶ Funkcja działania na zbiorze elementów



Przykład 5

ss > integration > e2e > JS basics4.spec.js > ...

```
/// <reference types="cypress" />
```

```
describe('User can google Wikipedia', function () {
  it('Queary and Url after googling should containt Wikipedia', function () {
    cy.visit("http://www.google.com");
    cy.viewport(1920,1080);
    cy.get('#L2AGLb > .QS5gu').should("be.visible");
    cy.get('#L2AGLb > .QS5gu').click();
    cy.get('#L2AGLb > .QS5gu').should("not.be.visible");
    cy.get("input[name='q']").clear().type('Wikipedia{enter}');
    cy.get("div[data-async-context='query:Wikipedia']").should('be.visible');
    cy.url().should('include','Wikipedia');
  })
  it('User should be able to go from search to page', function () {
    cy.viewport(1920,1080);
    cy.get("div[data-async-context='query:Wikipedia']").children().first().children().children()
      .first().next().children().first().find('a[href*="wikipedia.org/wiki/Wikipedia"]').eq(0).click();
    cy.url().should('include','https://pl.wikipedia.org/');
  })
})
```

Uruchamianie

- ▶ Cypress może pracować również bez okna z linii poleceń
- ▶ Na start mamy polecenie
 - ▶ `npx cypress run`
- ▶ Uruchomi wszystkie testy z konfiguracją zgodną z `cypress.json`
- ▶ Na koniec wykonania zwróci wynik w tabeli z informacją przechodzenia testów



Uruchamianie

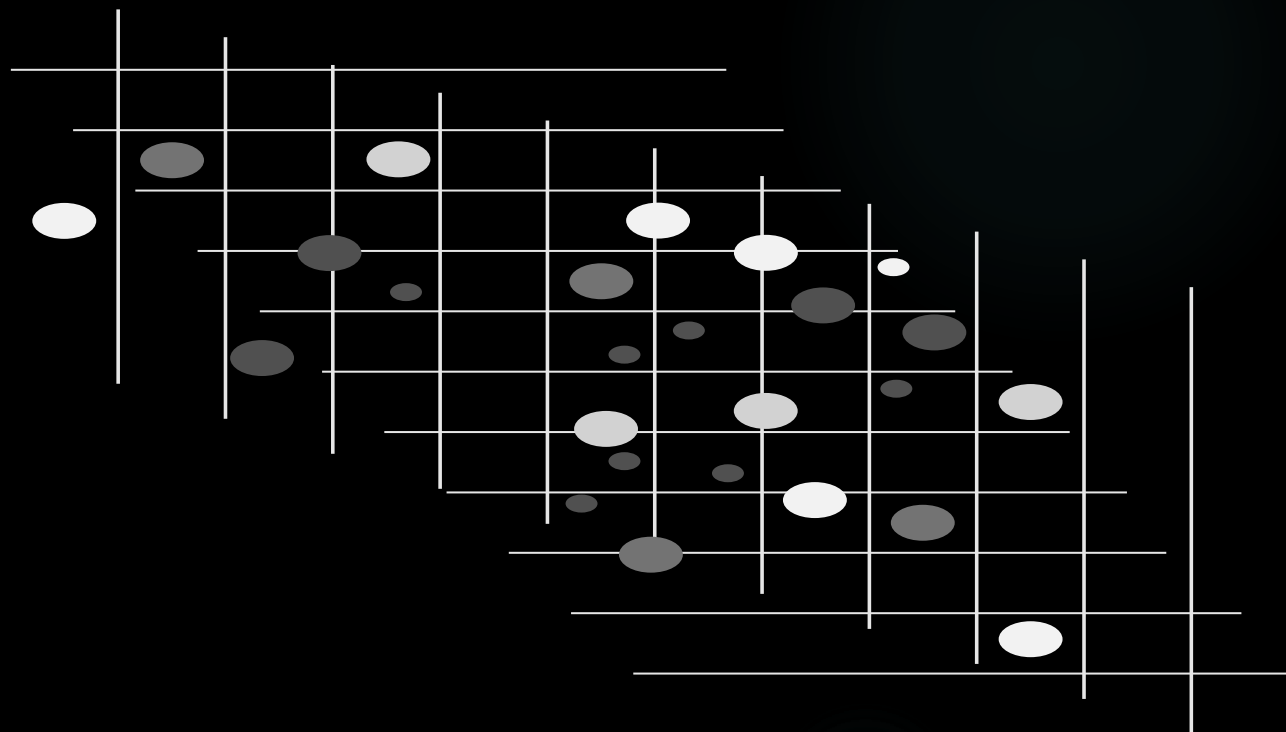
- ▶ Wynik zawiera
 - ▶ Nazwy plików
 - ▶ Czas trwania testu
 - ▶ Ilość testów
 - ▶ Ilość testów które przeszły
 - ▶ Ilość testów które nie przeszły
 - ▶ Ilość testów oczekujących
 - ▶ Ilość testów pominiętych

Spec		Tests	Passing	Failing	Pending	Skipped
✓ 1-getting-started/todo.spec.js	00:03	6	6	-	-	-
✓ 2-advanced-examples/actions.spec.js	00:14	14	14	-	-	-
✓ 2-advanced-examples/aliasing.spec.js	00:02	2	2	-	-	-
✓ 2-advanced-examples/assertions.spec.js	00:03	9	9	-	-	-
✓ 2-advanced-examples/connectors.spec.js	00:02	8	8	-	-	-
✓ 2-advanced-examples/cookies.spec.js	00:01	5	5	-	-	-
✓ 2-advanced-examples/cypress_api.spec.js	00:02	12	12	-	-	-
✓ 2-advanced-examples/files.spec.js	00:01	4	4	-	-	-
✓ 2-advanced-examples/local_storage.spec.js	00:01	1	1	-	-	-
✓ 2-advanced-examples/location.spec.js	00:01	3	3	-	-	-
✓ 2-advanced-examples/traversal.spec.js	00:03	18	18	-	-	-
✓ 2-advanced-examples/utilities.spec.js	00:03	5	5	-	-	-
✓ 2-advanced-examples/viewport.spec.js	00:03	1	1	-	-	-
✓ 2-advanced-examples/waiting.spec.js	00:06	2	2	-	-	-
✓ 2-advanced-examples/window.spec.js	00:01	3	3	-	-	-
✓ e2e/basics.spec.js	00:04	3	3	-	-	-
✓ e2e/basics2.spec.js	00:05	3	3	-	-	-
✓ e2e/basics3.spec.js	00:04	2	2	-	-	-
✓ e2e/basics4.spec.js	00:05	2	2	-	-	-
✓ e2e/firsttest.spec.js	00:06	1	1	-	-	-

Dodatkowe opcje



- ▶ Cypress sam zrzuci na dysk
 - ▶ Zrzuty Ekranu
 - ▶ Nagrania Ekranu
- ▶ Te zachowanie można zmienić przez plik `cypress.json`



Uruchamianie



- ▶ Mamy Również opcje uruchamiania pojedynczych plików
 - ▶ `npx cypress run --spec "adres pliku"`
- ▶ Również całego folderu
 - ▶ `npx cypress run --spec "adres folderu*"`

Przykład 6



```
=====
(Run Finished)

Spec                                Tests  Passing  Failing  Pending  Skipped
✓ e2e/basics.spec.js                00:04    3        3        -        -        -
✓ e2e/basics2.spec.js               00:04    3        3        -        -        -
✓ e2e/basics3.spec.js               00:03    2        2        -        -        -
✓ e2e/basics4.spec.js               00:05    2        2        -        -        -
✓ e2e/firsttest.spec.js             00:06    1        1        -        -        -
✓ All specs passed!                 00:25   11       11        -        -        -

PS D:\Workspaces\Cypress\Installation1> npx cypress run --spec ".\cypress\integration\e2e\*"

```

```
=====
(Run Finished)

Spec                                Tests  Passing  Failing  Pending  Skipped
✓ e2e/basics4.spec.js                00:08    2        2        -        -        -
✓ All specs passed!                 00:08    2        2        -        -        -

PS D:\Workspaces\Cypress\Installation1> npx cypress run --spec ".\cypress\integration\e2e\basics4.spec.js"

```