

Component Testing with Playwright



Debbie O'Brien

Senior Program Manager at Microsoft



Playwright Advocate

International Speaker

Teacher

YouTuber

Open Source

Love Sport: running, cycling, skiing, padel, 4th degree black belt in Taekwondo



Experts



debs_obrien

Node.js



Powerful
Test Runner



<button/>



<counter/>



<accordion/>

jsdom

Capable test
runner

- ☒ File System access
- ☒ Parallelization

Fake Browsers

- ☒ Fake CSS / layout
- ☒ No screenshot testing
- ☒ Fake clicks

Browser

Node.js

Browser



Light-weight
Test Runner

Poor test runner

- ✗ File System access
- ✗ Parallelization

Real Browsers

- ✓ Real CSS / layout
- ✓ Screenshot testing
- ✗ Fake clicks



<button/>



<counter/>




<accordion/>

Node.js

Browser




 Powerful
Test Runner


Powerful test runner


- ☒ File System access
- ☒ Parallelization

Real Browsers

- ☒ Real clicks
- ☒ Real CSS / layout
- ☒ Screenshot testing

 `<button/>`

 `<counter/>`

 `<accordion/>`

HelloWorld.spec.tsx

```
import { test, expect } from '@playwright/experimental-ct-vue'
import HelloWorld from './HelloWorld.vue'

test('should have a msg prop', async ({ mount }) => {
  const component = await mount(<HelloWorld msg='VueConf US' />)
  expect(component).toContainText('VueConf US')
})
```




HelloWorld.spec.tsx X



Filter (e.g. text, !...)

1/1 tests passed
(100%)

- src
 - components
 - HelloWorld.spe...
 - should have a...

```
1  import { test, expect } from "@playwright/experimental-ct-vue";
2  import HelloWorld from "../HelloWorld.vue";
3
4  test("should have a msg prop", async ({ mount }) => {
5    const component = await mount(<HelloWorld msg="VueConf US"></HelloWorld>);
6    await expect(component).toContainText("VueConf US");
7  });
8
```



0 0



Debbie



Live Share

UTF-8

LF



TypeScript React



Spell



Prettier



Formatting: ✓

```
import { test, expect } from '@playwright/experimental-ct-vue'
import WelcomeItem from './WelcomeItem.vue'

test('named slots should work', async ({ mount }) => {
  const component = await mount(WelcomeItem, {
    slots: {
      header: 'Header',
    }
  })
  await expect(component).toContainText('Header')
})
```



```
import { test, expect } from '@playwright/experimental-ct-vue'
import WelcomeItem from './WelcomeItem.vue'

test('named slots should work', async ({ mount }) => {
  const component = await mount(
    <WelcomeItem>
      <template v-slot:header>
        Header
      </template>
    </WelcomeItem>
  )
  await expect(component).toContainText('Header')
})
```

```
import { test, expect } from '@playwright/experimental-ct-vue'
import Button from './Button.vue'

test('event should work', async ({ mount }) => {
  const messages = []
  const component = await mount(Button, {
    props: {
      title: 'Submit'
    },
    on: {
      submit: data => messages.push(data)
    }
  })
  await component.click()
  expect(messages).toEqual(['hello'])
})
```

```
import { test, expect } from '@playwright/experimental-ct-vue'
import Button from './components/Button.vue'

test('event should work', async ({ mount }) => {
  const messages = []
  const component = await mount(
    <Button title='Submit'
      v-on:submit={data => {messages.push(data)}}>
    </Button>
  )
  await component.click()
  expect(messages).toEqual(['hello'])
})
```



Compile Component Bundle and Facade HTML Page

Under the Hood: Global Setup

<http://localhost:3100/playwright/index.html>
<http://localhost:3100/playwright/bundle.js>

```
HelloWorld.spec.tsx

import { test, expect } from '@playwright/experimental-ct-vue'
import HelloWorld from './HelloWorld.vue'

test('should have a msg prop', async ({ mount }) => {
  const component = await mount(<HelloWorld msg='VueConf US />)
  await expect(component).toContainText('VueConf US')
})
```

```
[
  { name: "HelloWorld", file: ... },
  { name: "WelcomeItem", file: ... },
  { name: "Button", file: ... },
]
```



Compile & Bundle

file:///...**cache**/playwright/index.html
file:///...**cache**/playwright/bundle.js



Under the Hood: Mount Fixture

```
import { test, expect } from '@playwright/experimental-ct-vue'
import HelloWorld from './HelloWorld.vue'

test('should have a msg prop', async ({ mount }) => {
  const component = await mount(<HelloWorld msg='VueConf US />)
  await expect(component).toContainText('VueConf US')
})
```

```
await page.goto('/playwright/index.html');

await page.evaluate(() => {
  window.bundleRender({
    name: 'HelloWorld',
    props: {
      msg: "VueConf US",
    },
    children: [],
  });
});
```



Component Tests Features

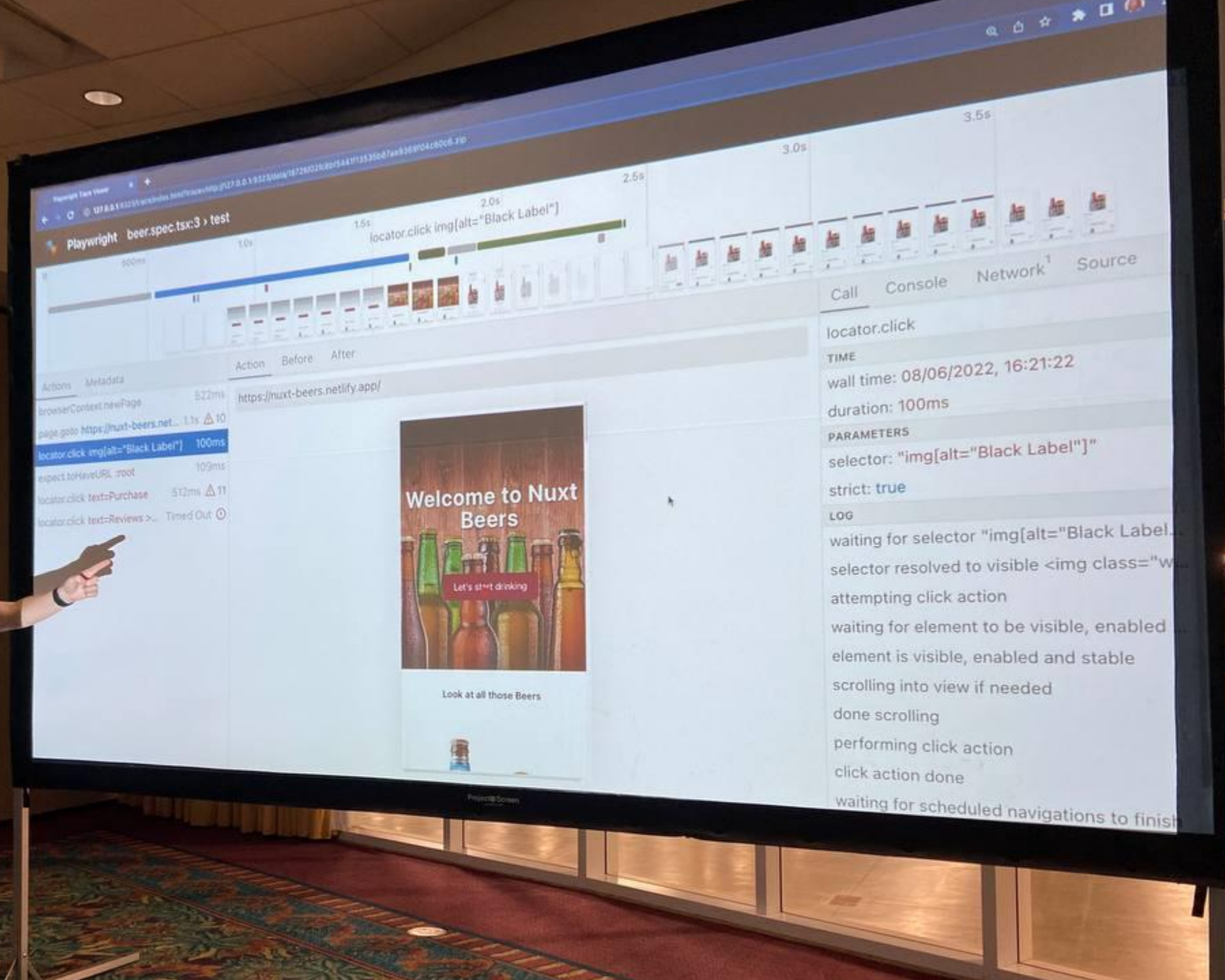
Real Browsers

- Chromium, Firefox, WebKit

All Playwright capabilities

Great Developer Experience

- importing components and using JSX syntax where possible
- handling component events in tests
- VSCode extension, debugging, Playwright Tracing etc.





Thank You

Docs: playwright.dev

aka.ms/playwright/twitter

aka.ms/playwright/youtube

aka.ms/playwright/slack