<https://adequatica.medium.com/advantages-and-disadvantages-between-cypress-and-playwright-6b29e3989108>

# Playwright

## Advantages:

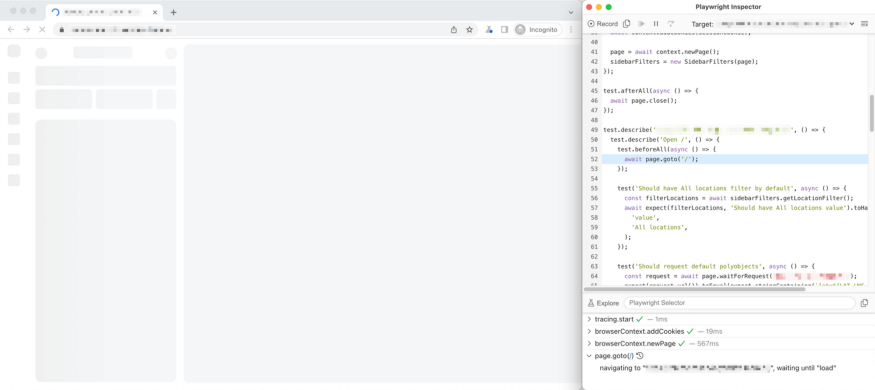
1. [Inspector](https://playwright.dev/docs/inspector) — a GUI tool for debugging tests. Not as good as Cypress has, but steel worthwhile.
2. Fast work in headless mode (I did not make accurate performance measurements, but it just feels faster than Cypress).
3. [Auto-waiting](https://playwright.dev/docs/actionability) of elements before performing some actions.
4. [Visual regression testing](https://playwright.dev/docs/test-snapshots) is working out of the box (in Cypress you have to use a plugin).
5. Detailed HTML report, which could be useful both at local run and in CI.
6. Auto-making [screenshots](https://playwright.dev/docs/test-configuration" \l "automatic-screenshots" \t "_blank), [videos](https://playwright.dev/docs/test-configuration" \l "record-video" \t "_blank) and traces on failures — all of them are great for debugging failed tests, especially in [Trace Viewer](https://playwright.dev/docs/trace-viewer).

## Curious or controversial features:

1. [Test generator](https://playwright.dev/docs/codegen) — a GUI tool for record tests. Of course, the generated results can not be used for real tests: chosen selectors are odd and need to be replaced, and expects should be added manually later. But it could be handy for creating rough drafts for simple tests.
2. There is no support for custom commands, as Cypress and WebdriverIO have. Playwright pushes you to use [Page Object Model](https://playwright.dev/docs/test-pom) instead of creating custom syntactic sugar around your testing infrastructure.
3. Most of [page](https://playwright.dev/docs/api/class-page" \t "_blank)’s and [locator](https://playwright.dev/docs/locators" \t "_blank)’s methods work only with the first element on the page. You have to seek out special methods which are allowed to work with multiple elements. Playwright pushes you to choose extremely unambiguous selectors, while Cypress or WebdriverIO do not limit you this way.
4. P̶l̶a̶y̶w̶r̶i̶g̶h̶t̶ ̶d̶o̶e̶s̶ ̶n̶o̶t̶ ̶a̶l̶l̶o̶w̶ ̶p̶a̶s̶s̶i̶n̶g̶ ̶a̶n̶ ̶a̶r̶b̶i̶t̶r̶a̶r̶y̶ ̶h̶e̶a̶d̶e̶r̶ ̶t̶o̶ ̶t̶h̶e̶ ̶t̶e̶s̶t̶ ̶p̶a̶g̶e̶ (sorry, I figured it out, [it is possible](https://playwright.dev/docs/api/class-page" \l "page-set-extra-http-headers" \t "_blank)). But anyway, I could not pass a single header Cookies with multiple cookies, I had to [set each cookie](https://playwright.dev/docs/api/class-browsercontext#browser-context-add-cookies) one by one. The same strange thing with localStorage: I could not pass only a single localStorage to [the new browser’s context](https://playwright.dev/docs/api/class-browser#browser-new-context), because setting a storage state implies the object of both mandatory cookies and localStorage (and not just one thing). On the one hand, this is strictly right, but on the other hand it is an excessive restriction.

## Disadvantages:

1. Сonfusing documentation. Without a doubt a Playwright’s documentation is good, but looking for lore takes time and examples are quite poor. After the [installation process](https://playwright.dev/docs/intro) I had to use Google and Stackoverflow to figure out some of the details.
2. Despite the fact that Playwright allows [API testing](https://playwright.dev/docs/test-api-testing), its [request method](https://playwright.dev/docs/api/class-apirequest" \t "_blank) is not enough advanced. For example, it does not support disabling follow redirects ([but Cypress does](https://docs.cypress.io/api/commands/request)).
3. Screenshots, videos and traces on failures are made inside one [test()](https://playwright.dev/docs/api/class-test) function. It means that if you have a few tests inside test.describe(), you will get videos and traces for each nested test, but not for a whole root test function ([steps](https://playwright.dev/docs/api/class-test" \l "test-step" \t "_blank) have the same behavior). It is a shame that so many artifacts can not be assembled into one piece.
4. TeamCity is not supported out of the box — you have to write your own reporter for CI builds.



Playwright in Debug Mode

Playwright is a more technical tool, which works closer to the browser and has precise commands. Cypress is a more flexible tool with an expectable API for testers, which allows to develop tests easy and fast enough.

<https://www.browserstack.com/guide/playwright-vs-cypress>

Une image contenant table

Description générée automatiquement