



Automation of GUI testing using visual regression testing

Audun Wigum Arbo

Problem: How to test that something is visually correct?

```
▼<div class="jsx-87433190 page-transition">
  ▼<div class="jsx-1126205319 new__home__container">
    ▼<div class="jsx-661675692 main__header">
      ▼<div class="jsx-661675692 main__header-center">
        <span class="jsx-661675692 main__header-title">NDC
        Oslo</span>
        <br class="jsx-661675692">
        "22-26 May 2023"
        <br class="jsx-661675692">
        "Oslo Spektrum"
        <br class="jsx-661675692">
      </div>
    </div>
  ><div class="jsx-3601836161 css-14ml1ci">...</div>
  ><div class="jsx-1893824686 blocks__speakers__containe
  r">...</div> flex
```



Visual (regression) testing

 **percy** +  **Playwright**

What is visual testing?

1. Test using visual snapshots – not asserts
2. Capture snapshots in the same place every test run
3. Compare old and new snapshots
4. Visual changes? Notify!

A simple regression test

Does NDC's home page look correct?

Write an end-to-end test in Playwright

```
import { test, expect } from "@playwright/test";
import percySnapshot from "@percy/playwright";

test("NDC homepage looks good", async ({ page }) => {
  // Wait until page is fully loaded
  await page.goto("https://ndcoslo.com/", { waitUntil: "networkidle" });
  // Take a snapshot and send to Percy
  await percySnapshot(page, "NDC home page");
});
```

Check snapshot in Percy

1/1 NDC home page

```
await percySnapshot(page, "NDC home page");
```

New snapshot

Changes from main
a minute ago

NDC { Oslo }

Tickets Agenda Workshops Speakers


Menu



Tickets

NDC Oslo



22-26 May 2023


Someone makes a change and Percy complains!





AudunWA commented 7 days ago  

No description provided.

 Change some details on the website ✗ 8b4ab4e



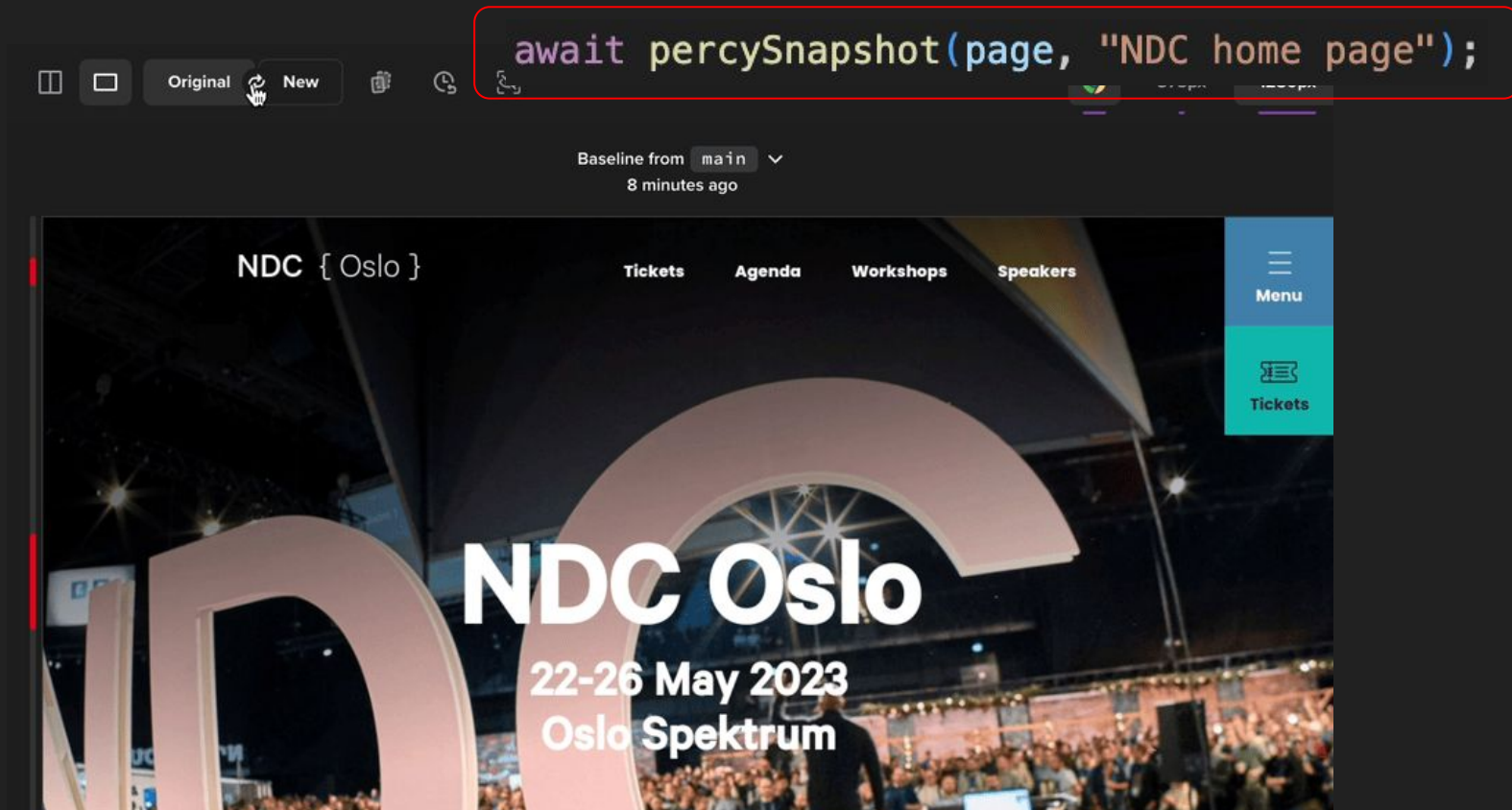
 **Some checks were not successful** [Hide all checks](#)
1 successful and 1 errored checks

✗  **percy** — 1 visual change needs review [Details](#)

Merge pull request ▼




You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Percy finds differences between new and old snapshot



What more can we do?

Some tips to take visual testing to the next level!

- Write similar tests for all important pages
 -  A “visual smoke test”
- Test with other browsers, devices and screen sizes
 -  Simple changes in Percy config
- Integrate with CI and GitHub
 -  Faster reviews, fewer regressions

3 common challenges

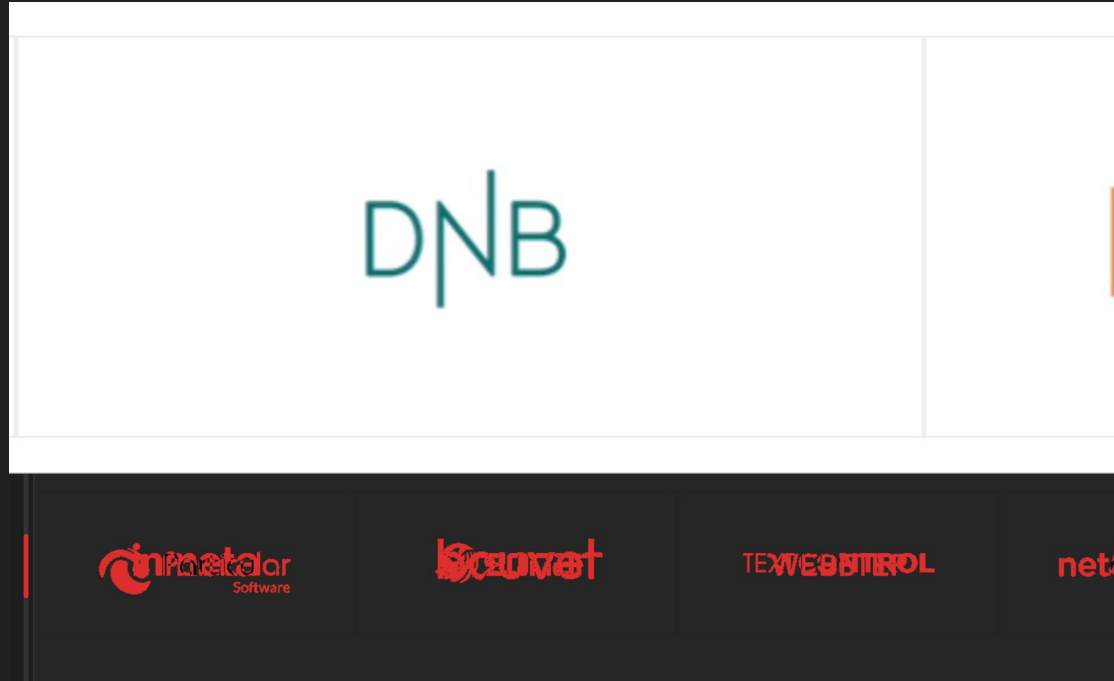
They often appear,
but we can fix them!





Periode	Oppdatert
-	5. okt. 2022 BUDS.

- ✗ Live dates (changes every day)
- ✗ Not waiting for completed loading
- ✗ Live environment with changes

Troubles with NDC page



In summary

- Testing GUI code is tedious
- Visual comparison of snapshots ( **percy**)
in e2e tests ( **Playwright**)
- Quickly make value with “visual smoke testing”

Thank you!

linkedin.com/in/audunwa

audun.me