

Website Testing mit Cypress

Erfolgreiche und zuverlässige E2E-Tests mit Cypress
Wie fange ich an?

Contao Konferenz 2023



pdir / digital agentur

Mathias Arzberger

2

- » Geboren 1982
- » Seit 15 Jahren Contao im Einsatz
- » Geschäftsführer der pdir GmbH
- » Themes, Erweiterungen, Shops
- » Wartungsverträge für Agenturen und Endkunden

- » <https://pdir.de>
- » <https://contao-themes.net>
- » <https://maklermodul.de>

Agenda

- » Was ist Cypress?
- » Warum sollte ich eine Website testen?
- » Welche Fehler kann ich durch Tests beheben?
- » Wer hat vor Cypress die Website getestet?
- » Welche Browser unterstützt Cypress?
- » Cypress installieren
- » Den ersten Test schreiben
- » Test laufen lassen
- » Was testet pdir in einem Kundenprojekt
- » Cypress Chrome Recorder
- » Cypress und Lighthouse

- » Level: Grundkenntnisse in JavaScript erforderlich

Was ist Cypress?

4



- » End-to-End-Testframework für Web-Anwendungen, das auf JavaScript basiert und in Node.js ausgeführt wird.

Leistungsstark dank

- » **In-Browser-Testausführung:** Cypress führt Tests direkt im Browser aus, sodass Sie die Benutzer:innenerfahrung aus erster Hand testen können.
- » **Ausführliche Fehlermeldungen:** Cypress generiert ausführliche Fehlermeldungen, die dir helfen, Fehler zu finden und zu beheben.
- » **Automatisierte Screenshot-Erstellung:** Cypress erstellt automatisch Screenshots von fehlgeschlagenen Tests, um bei der Fehlerbehebung zu helfen.
- » **Gute Integration mit IDEs:** Cypress bietet eine gute Integration mit IDEs wie Visual Studio Code, PhpStorm und IntelliJ IDEA.

Warum sollte ich eine Website testen?

5

- » Vermeidung von Fehlern und Problemen
 - » Besucherzahlen, Konversionsraten
- » Optimierung des Nutzererlebnisses
 - » A/B Testing, was bevorzugen Besucher:innen?
- » Verbesserung der SEO
 - » Meta-Tags, URL-Struktur, Indexierung
- » Reduzierung von Risiken
 - » Schlechten Ruf und Rechtsstreitigkeiten vermeiden
- » Nachweis für beide Seiten
 - » Bedingungen der Auftragsvergabe prüfen
 - » Nachweis DSGVO

- » Fazit

Durch das Testen einer Website kann man diese Probleme identifizieren und beheben, bevor sie den Besucher:innen der Website begegnen. Dies trägt dazu bei, die Qualität der Website zu verbessern und die Nutzererfahrung zu optimieren.

Welche Fehler kann ich durch Tests beheben?

- » Technische Probleme
- » Usability-Probleme
- » Design-Probleme
- » Suchmaschinen-Indexierungsprobleme
- » Rechtliche Probleme

Welche Fehler kann ich durch Tests beheben?

» Technische Probleme

- » Fehler in der Programmierung aufdecken
- » Probleme der Ladegeschwindigkeit
- » Inkompatibilitäten mit Browser oder Geräten
- » Testen der Leistungsfähigkeit mit Google Lighthouse

» Usability-Probleme

- » Navigation
- » Bedienbarkeit
- » Verständlichkeit der Inhalte

Welche Fehler kann ich durch Tests beheben?

- » Design-Probleme
 - » Ästhetik
 - » Lesbarkeit
 - » Konsistenz des Designs

- » Suchmaschinen-Indexierungsprobleme
 - » Auffinden der sitemap.xml und robots.txt
 - » Weiterleitungen
 - » Indexierbare Seiten

Welche Fehler kann ich durch Tests beheben?

9

» Rechtliche Probleme

- » Pflichtangaben für bestimmte Branchen
- » 2-Klick-Regel des BGB
- » Consent für Cookies
- » Datenschutzhinweise für externe Dienst

Wer hat vor Cypress die Website getestet?

10

Oder besser gesagt, wer hat die Fehler vorher gefunden?

Wer hat vor Cypress die Website getestet?

11

Oder besser gesagt, wer hat die Fehler vorher gefunden?

- » Im besten Fall mindestens ein pdir Mitarbeiter:innen bei der Entwicklung

Viel wahrscheinlicher:

- » Der Auftraggeber:innen
- » Die Besucher:innen der Website

Welche Browser werden von Cypress unterstützt?

12

- Chrome
- Chrome Beta
- Chrome Canary
- Chromium
- Edge
- Edge Beta
- Edge Canary
- Edge Dev
- Electron
- Firefox
- Firefox Developer Edition
- Firefox Nightly
- WebKit (Experimental)

Cypress installieren

- » Node.js ist bereits installiert.
- » Na dann los!

```
npm init
```

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

```
npx cypress open
```

Den ersten Test schreiben

- » Cypress-Tests werden in JavaScript geschrieben und in Dateien mit der Endung `.cy.js` gespeichert.

```
describe("Öffne meine Website", () => {  
  it("Sollte eine Fehlermeldung anzeigen", () => {  
    // Öffnen der Website  
    cy.visit("/");  
  });  
});
```

Test laufen lassen

```
npx run cypress open
```

- » Anpassen der package.json für kürzere Aufrufe von Cypress auf der Konsole

```
{  
  "scripts": {  
    "cy:open": "cypress open"  
    "cy:run": "cypress run"  
  }  
}
```

- » Öffne die grafische Oberfläche

```
npm run cy:open
```

- » Führe die Tests in der Kommandozeile aus

```
npm run cy:run
```

Was testet pdir in einem Kundenprojekt?

16

» Was testen wir?

Was testet pdir in einem Kundenprojekt?

17

- » Aufruf der Website
- » Enthält das title- oder description-Tag bestimmte Wörter
- » Prüfen auf Cookies und Auflisten der Cookies
- » Weiterleitung SSL / www / kein www
- » Weiterleitung von Kurz-Url / QR-Codes
- » Sind Debug- oder Fehlermeldungen vorhanden
- » Sind bestimmte Unterseiten im Menü verlinkt
- » Prüfung von Formularen
- » Werden Formulardaten im Backend erfasst
- » Aufrufen aller Unterseiten, finden von 404 Links usw.
- » Kann die robots.txt und die sitemap.xml fehlerfrei geladen werden und sind bestimmte Inhalte vorhanden
- » Werden externe Scripte geladen (CDN, Tracking, iframe)
- » Google Lighthouse Tests
- » Prüfen aller Funktionen des Consent Manager
- » Prüfen aller Funktionen des Google Tag Manager
- » Auf valides HTML prüfen

Cypress Chrome Recorder

18

» Tests aufnehmen

» Chrome Recorder + Erweiterung + Export für Cypress
oder

» Installation

```
npm init -g @cypress/chrome-recorder
```

» Tests für Cypress umwandeln

```
npx @cypress/chrome-recorder
```

» Links

- » <https://chrome.google.com/webstore/detail/cypress-chrome-recorder/fellcphjglholofndfmmjmheedhomgin>
- » <https://github.com/cypress-io/cypress-chrome-recorder>

Cypress, Lighthouse und Pa11y

19

» Was ist Google Lighthouse?

» Installation

```
npm i -D cypress @cypress-audit/lighthouse
```

» Anpassen der commands.js

```
import "@cypress-audit/lighthouse/commands";
```

» Ein Beispiel

Danke.

» Noch Fragen?

» Wichtige Links zum Vortrag

Cypress Dokumentation

<https://docs.cypress.io/guides/getting-started/installing-cypress>

Cypress Lighthouse

<https://mfrachet.github.io/cypress-audit/start-with-why.html>

Pa11y

<https://pa11y.org/> bzw. <https://github.com/pa11y/pa11y>

Node.js

<https://nodejs.org/>