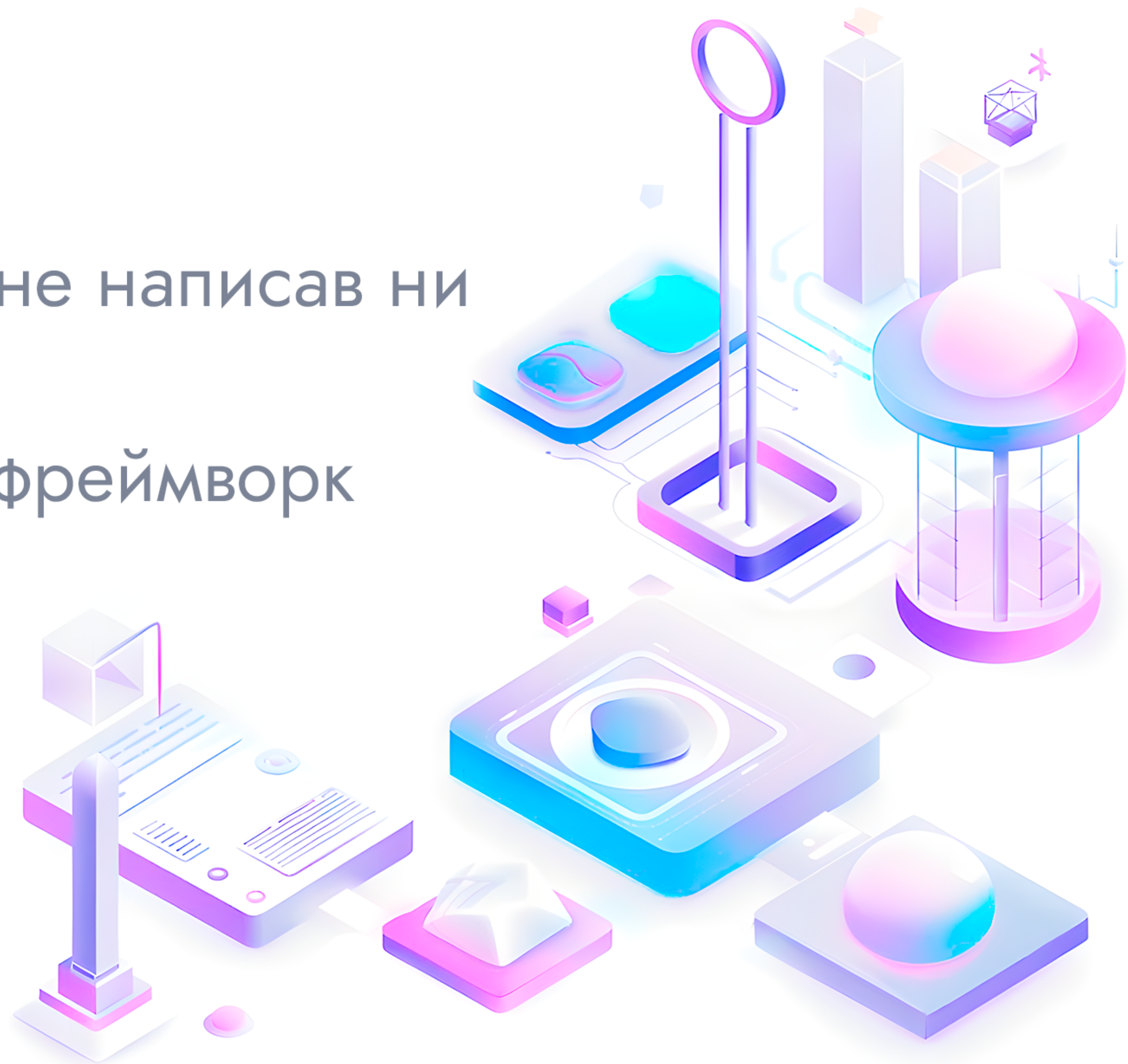
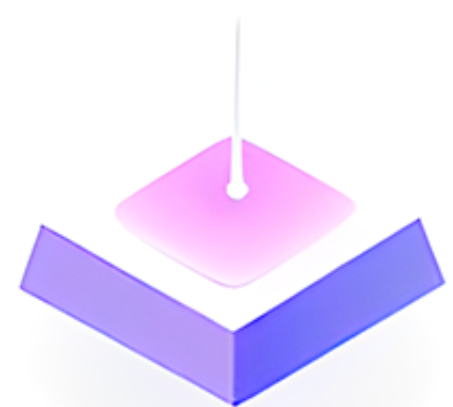


Testplane

Как проверить весь UI, не написав ни одного теста?

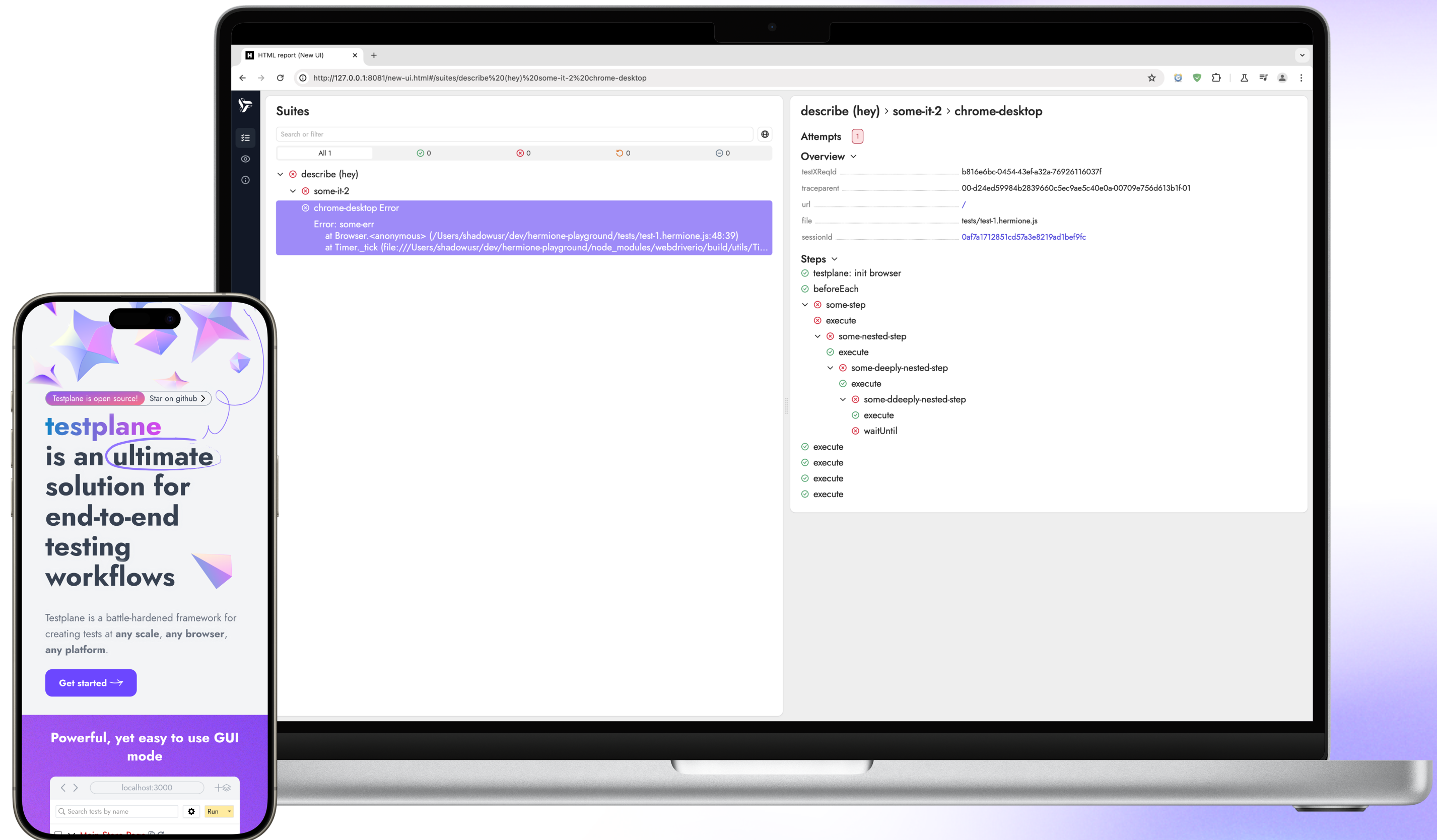
Как мы разрабатываем фреймворк для тестирования всего в вебе

Николай Марков



О себе

- ✓ 3 года занимаюсь развитием инфраструктуры end-to-end тестов в Яндексе
- ✓ Рисую и разрабатываю интерфейсы



Testplane

Универсальный фреймворк для тестирования интерфейсов

- ✓ Масштабируемость и скорость выполнения тестов
- ✓ Кросс-платформенность, кросс-браузерность и единое API
- ✓ Визуальные проверки и no-code тесты при использовании `storybook`
- ✓ Легкая расширяемость

Демо

Начало работы

- ✓ Достаточно воспользоваться одной командой в CLI для настройки проекта:

```
npm init testplane@latest
```

```
✓ Testplane and plugins have been installed successfully at /Users/shadowusr/dev/testplane-demo
```

```
Inside that directory, you can run:
```

```
npx testplane  
Runs testplane tests
```

```
npx testplane gui  
Launches testplane's gui
```

```
✓ TESTPLANE-DEMO  
  > node_modules  
  ✓ testplane-tests  
    TS example.testplane.ts  
    TS tsconfig.json  
  TS .testplane.conf.ts  
  {} package-lock.json  
  {} package.json
```



<https://testplane.io/ru/docs/v8/quickstart/>



Видео: установка Testplane

<https://disk.yandex.ru/i/c8GrJjrBmZS1EA>

Запуск тестов

- ✓ Можно запускать в CLI с помощью команды `npm test`
- ✓ Можно запускать с помощью кнопок в VS Code с помощью расширения
- ✓ Можно запустить Testplane UI и работать с тестами в нём



Видео: запуск тестов

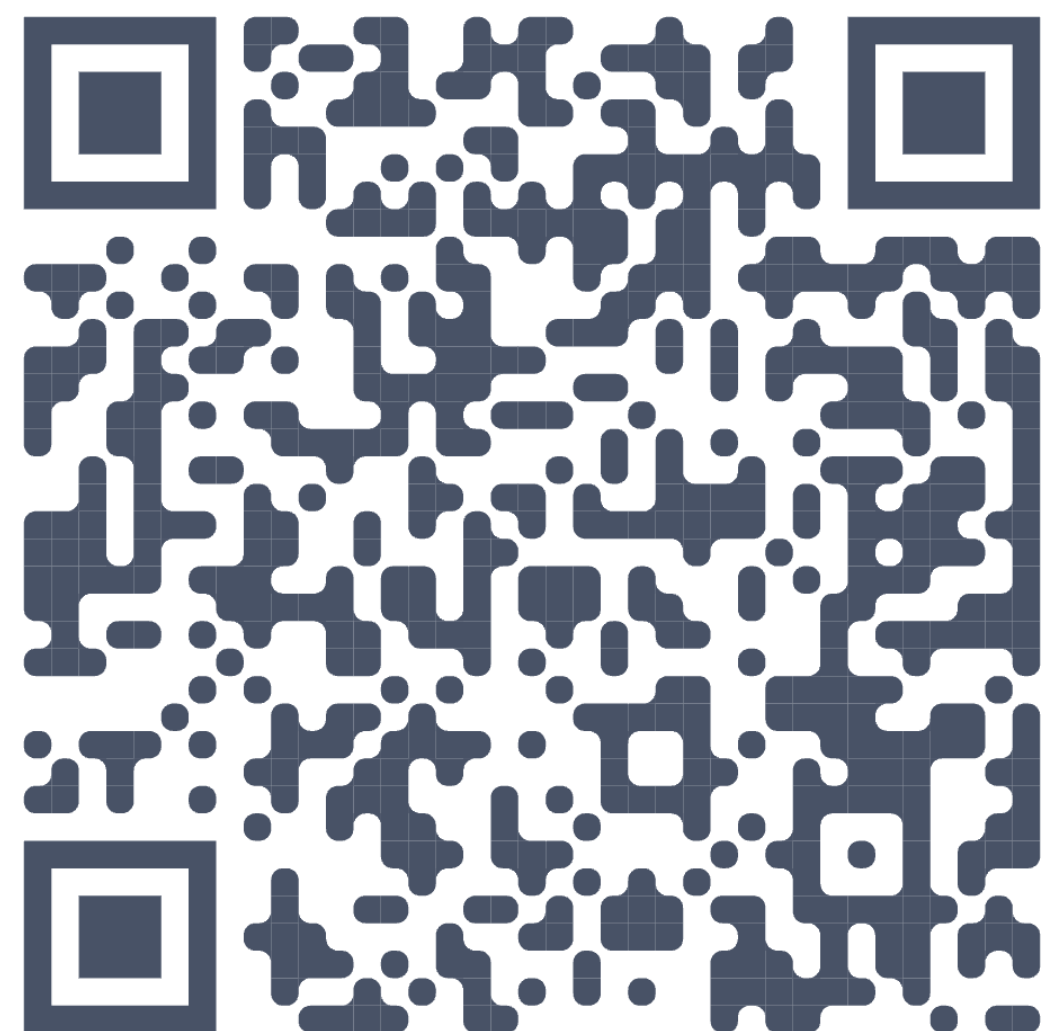
<https://disk.yandex.ru/i/1HonV9DqbCf2pw>

Разработка тестов

- ✓ Мгновенный отклик при написании тестов (REPL)
- ✓ Огромное разнообразие доступных команд
- ✓ Гибкость в выборе библиотеки assert'ов и локаторов



<https://testplane.io/ru/docs/v8/commands/browser/switchToRepl/>



Видео: разработка тестов и REPL

<https://disk.yandex.ru/i/sewE2XwNj1K5-g>

Testplane UI

- ✓ Удобный запуск тестов и просмотр результатов
- ✓ Полный флоу по работе с визуальными проверками
- ✓ Единый UI для отчетов в CI и локальной разработки



<https://testplane.io/ru/docs/v8/html-reporter/html-reporter-setup/>



Видео: Testplane UI

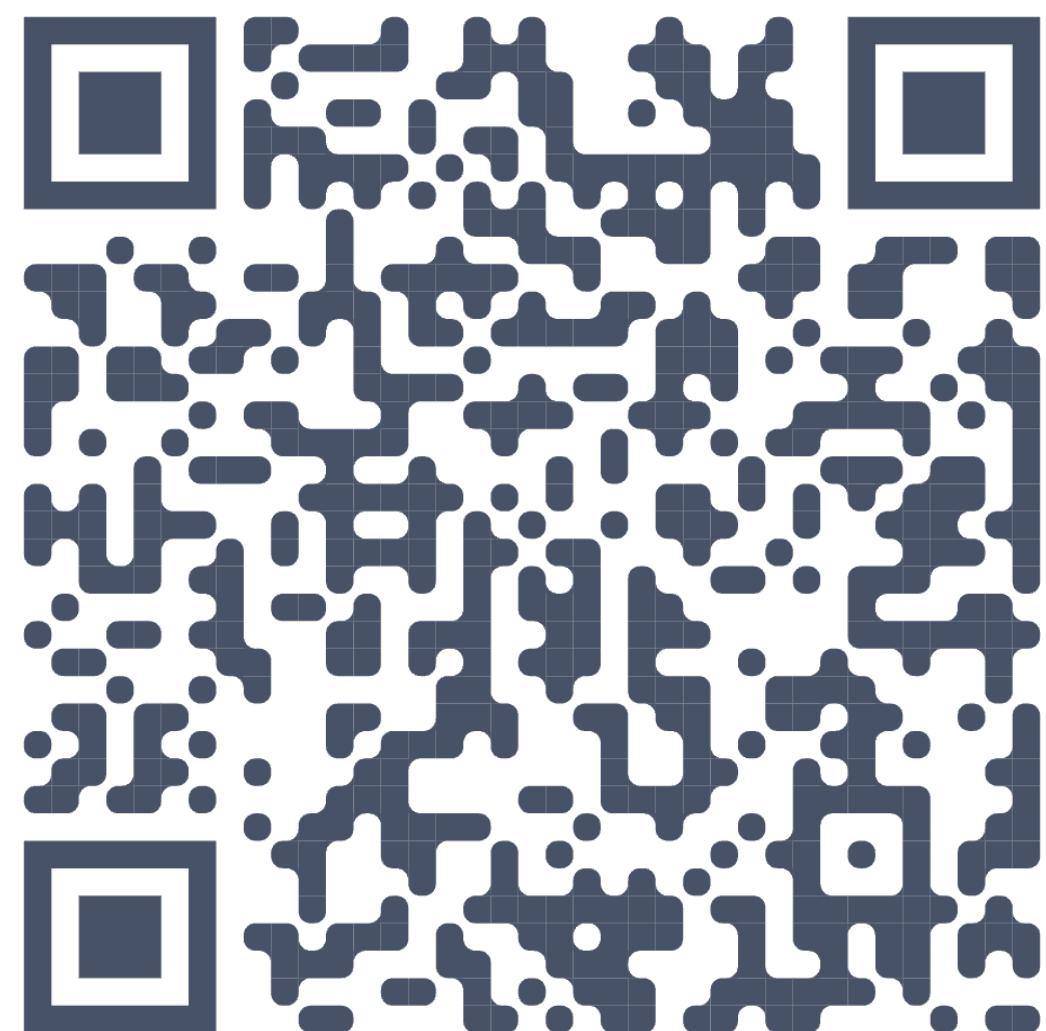
https://disk.yandex.ru/i/e35Gwo_AfMjVjg

Визуальные проверки

- ✓ Кастомные алгоритмы сравнения скриншотов
- ✓ Автоматическое ожидание и отключение анимаций
- ✓ Продуманный флоу работы со скриншотами



<https://youtu.be/crbwyGlcXm0>



Видео: визуальные проверки

<https://disk.yandex.ru/i/9stuNjg-DFFf1w>

Интеграция со Storybook

- ✓ Визуальные проверки всех компонентов без доп. кода
- ✓ Поддержка play-функций из Storybook
- ✓ Поддержка кастомных тестов в дополнение к play-функциям



<https://github.com/gemini-testing/testplane-storybook>



Видео: интеграция с Storybook

https://disk.yandex.ru/i/-TAf_j0HLfj8zg

Компонентное тестирование

- ✓ Возможность удобной отладки тестов и гарантия корректности в отличие от JSDom
- ✓ Есть Hot Module Replacement
- ✓ Тесты запускаются в браузером окружении (можно писать даже unit-тесты!)



<https://testplane.io/ru/blog/component-testing-intro/>

```

tests > example.testplane.tsx > it("test") callback
1
2 import { render } from "@testing-library/react";
3 import { Component } from "./component";
4
5 it("test", async ({ browser }) => {
6   render(<Component />);
7
8   await browser.$('button').click();
9   await browser.$('button').click();
10
11   await browser.switchToRepl();
12 });
13

```

PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

```

~/j/p/component-testing >>> ./node_modules/.bin/testplane --repl
[00:57:17 +0300] Vite server started on http://localhost:4001/
(node:80814) ExperimentalWarning: The Fetch API is an experimental feature. This feature could change at any time
(Use `node --trace-warnings ...` to show where the warning was created)
(node:80899) ExperimentalWarning: The Fetch API is an experimental feature. This feature could change at any time
(Use `node --trace-warnings ...` to show where the warning was created)
[00:57:20 +0300] You have entered to REPL mode via terminal (test execution timeout is disabled).
> [vite] hot updated: /tests/component.tsx
[00:58:03 +0300] You have entered to REPL mode via terminal (test execution timeout is disabled).
>

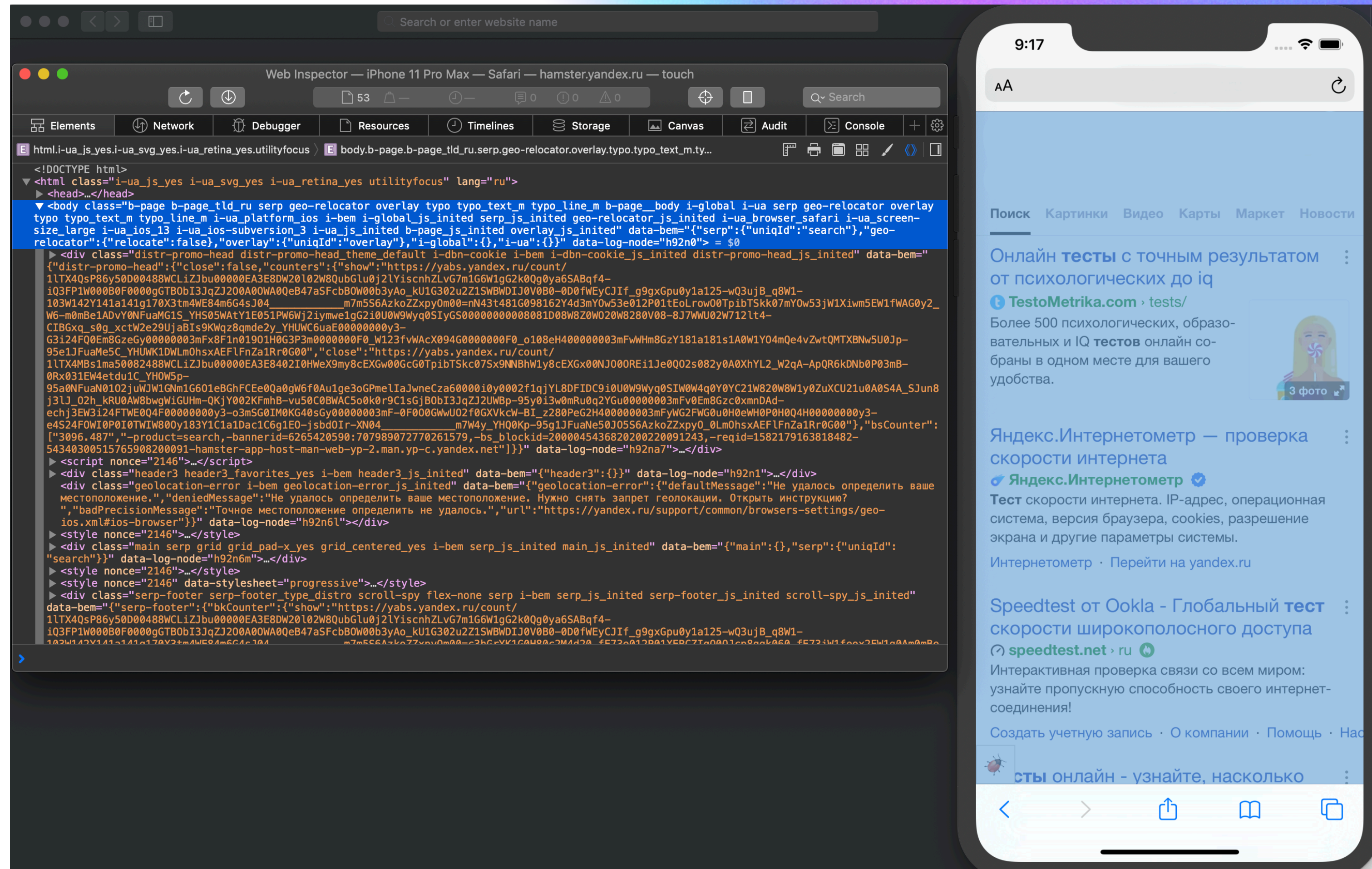
```

Chrome is being controlled by automated test software.

counter = 1

Любые платформы и браузеры

- ✓ Поддержка всех современных браузеров
- ✓ Удаленный грид
- ✓ Реальные устройства iOS/Android



Оптимизирован для CI

- ✓ Шардирование
- ✓ Единый UI в CI и локальной разработке
- ✓ Подробные логи и разнообразие отчетов
- ✓ Трейсинг всех запросов

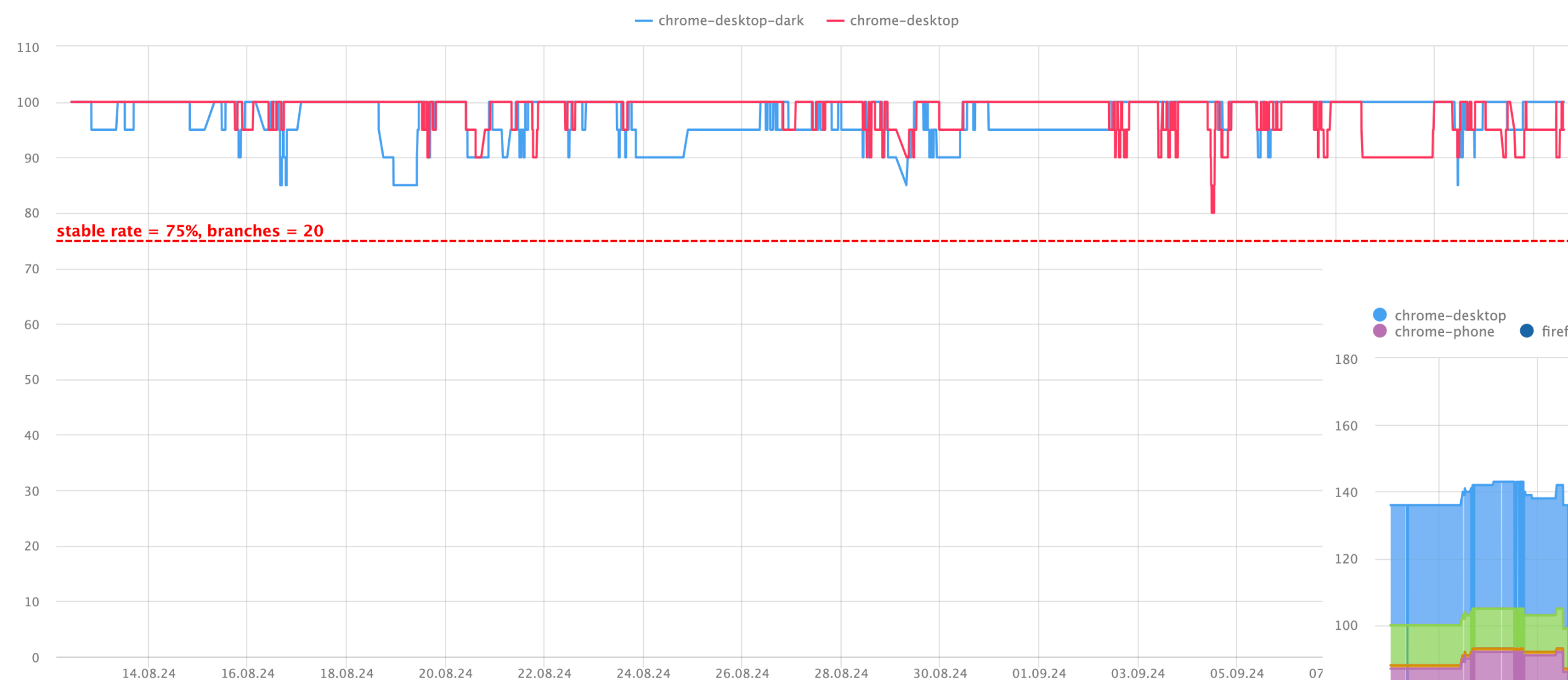
Простая кастомизация

- ✓ Легко подключить или реализовать свой отчет
- ✓ Можно писать свои команды
- ✓ Есть Programmatic API
- ✓ Десятки open-source плагинов

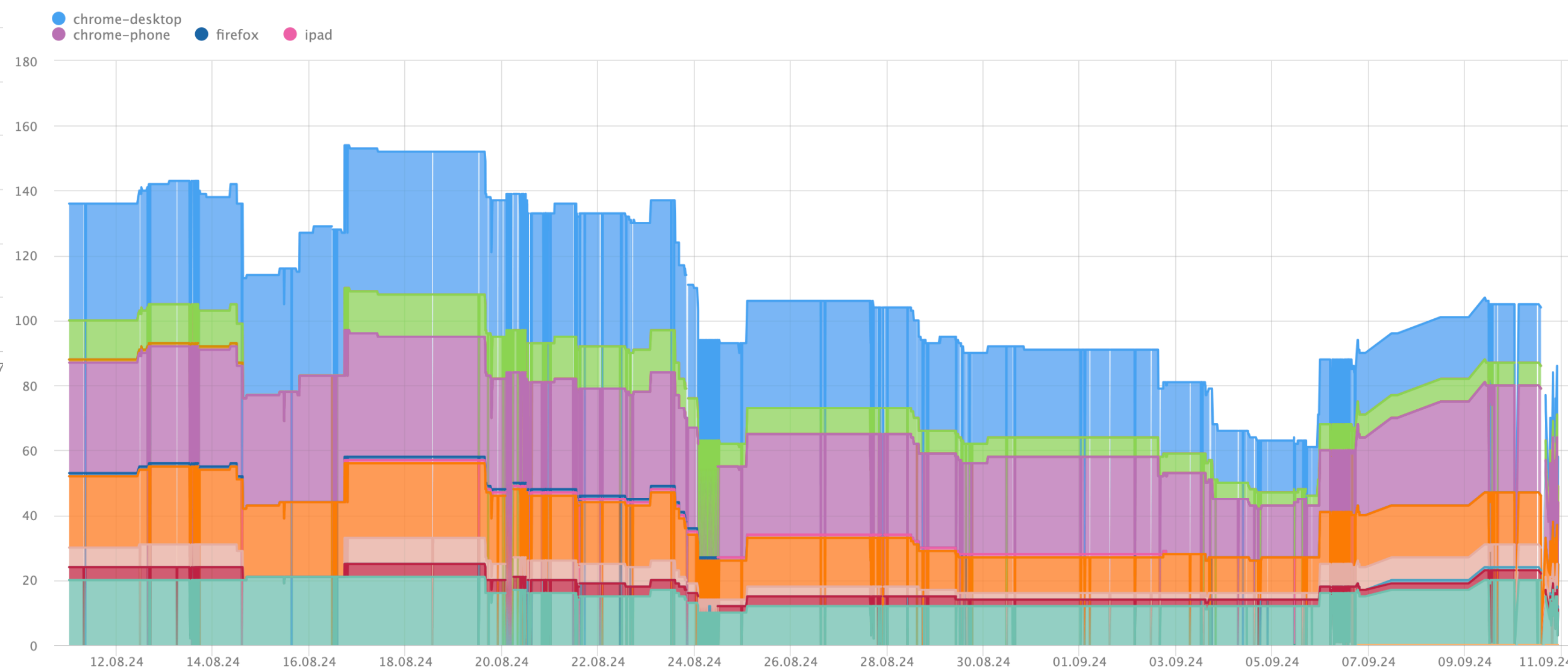
Пример кастомизации

Построение системы отслеживания стабильности тестов и их заскипа с помощью плагина к Testplane

2024-09-11 14:41:00 [project / testplane] Стабильность теста «Test name»



2024-09-11 14:41:00 [project / testplane] Количество заскипанных тестов во всех браузерах



Recap

Универсальный фреймворк для тестирования веба

- ✓ Мгновенный отклик при написании тестов
- ✓ Интуитивность в тестах: изоляция, автоматическое ожидание, ретраи
- ✓ Произвольные браузеры и окружения
- ✓ Мощный UI для работы с тестами и визуальными проверками
- ✓ Возможности по расширению

Testplane — open source

- ✓ Нет vendor lock-in
- ✓ MIT license
- ✓ Растущее комьюнити

Спасибо!

- ✓ Будем рады видеть вас на testplane.io
- ✓ Приходите с вопросами и фидбеком в community-чат t.me/testplane
- ✓ Ставьте звездочки на GitHub github.com/gemini-testing/testplane



testplane.io