

CodeceptJS

Support Us

Q Search...

Quickstart

Guides

Helpers

Plugins

API

Releases

Community

ua CodeceptJS was created in Ukraine. #StandWithUkraine

Web Testing

Getting Started

Architecture

Writing Tests

Opening a Page

Locating Element

Clicking

Filling Fields

Assertions

Grabbing

Waiting

How It Works

Running Tests

Level of Detail

Filter

Parallel Run

Configuration

Debug

Pause

Pause on Fail

Screenshot on Failure

Step By Step Report

Before

Locating Element

Element can be found by CSS or XPath locators.

```
I.seeElement('.user') // element with CSS class user
I.seeElement('//button[contains(., "press me")]') // button
```

By default CodeceptJS tries to guess the locator type. In order to specify the exact locator type you can pass an object called **strict locator**.

```
I.seeElement({ css: 'div.user' })
I.seeElement({ xpath: '//div[@class=user]' })
```

Strict locators allow to specify additional locator types:

```
// locate form element by name
I.seeElement({ name: 'password' })
// locate element by React component and props
I.seeElement({ react: 'user-profile', props: { name: 'daveit' } })
```

In *mobile testing* you can use `id` to specify the accessibility id to locate an element. In web application you can locate elements by their `aria-label` value.

Videos

Books & Posts

Examples

Cookbook

Commercial Services

Trainings

Testomat.io

Support us via OpenCollective!

CodeceptJS

Support Us

Q Search...

Quickstart

Guides

Helpers

Plugins

API

Releases

Community

ua CodeceptJS was created in Ukraine. #StandWithUkraine

Web Testing

Getting Started

Architecture

Writing Tests

Opening a Page

Locating Element

Clicking

Filling Fields

Assertions

Grabbing

Waiting

How It Works

Running Tests

Level of Detail

Filter

Parallel Run

Configuration

Debug

Pause

Pause on Fail

Screenshot on Failure

Step By Step Report

Before

BeforeSuite

Retries

Architecture

CodeceptJS bypasses execution commands to helpers. Depending on the helper enabled, your tests will be executed differently.

The following is a diagram of the CodeceptJS architecture:

```
graph TD
    CodeceptJS[CodeceptJS] --> Playwright[Playwright]
    CodeceptJS --> Puppeteer[Puppeteer]
    CodeceptJS --> WebDriver[WebDriver]
    CodeceptJS --> TestCafe[TestCafe]
    CodeceptJS --> Other[Other Helpers]
    
    Playwright --> CDP[CDP]
    Puppeteer --> CDP
    WebDriver --> Selenium[Selenium WebDriver]
    TestCafe --> Proxy[Proxy Server]
    
    CDP --> Safari[Safari]
    CDP --> Electron[Electron]
    CDP --> Chrome[Chrome]
    Selenium --> Firefox[Firefox]
    Selenium --> Edge[Edge]
    Proxy --> Chrome
    Proxy --> Firefox
    Proxy --> Edge
```

Videos

Books & Posts

Examples

Cookbook

Commercial Services

Trainings

Testomat.io

Support us via OpenCollective!