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: 'davert' } })
```

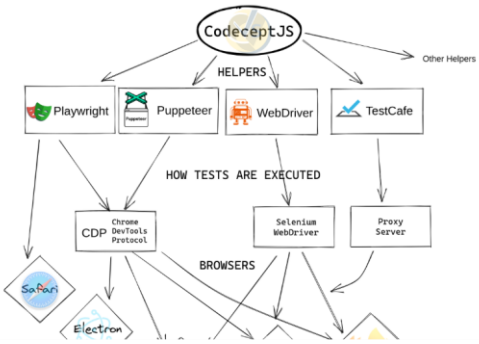
In mobile testing you can use `~` to specify the accessibility id to locate an element. In web application you can locate elements by their `aria-label` value.

CodeceptJS
Support Us
Search...
Quickstart
Guides
Helpers
Plugins
API
Releases
Community
UA CodeceptJS was created in Ukraine. #StandWithUkraine

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:



Videos
Books & Posts
Examples
Cookbook →
Commercial Services →
Trainings →
Testomat.io →
Plan your end 2 end tests, collaborate, synchronize with code & get reports! Join Testomat.io while it is in beta and get a huge discount!
Support us via OpenCollective!