Table of Contents ›

Testing

How to test your Nuxt application.

> 👉 Test utils are still in development and the API and behavior may change. Currently, it is in preview stage but not yet ready for testing production apps. If you are a module author, you can find more specific information in the Module Author's guide

In Nuxt 3, we have a rewritten version of `@nuxt/test-utils`. We support Vitest and Jest as test runners.

Installation

```
> yarn add --dev @nuxt/test-utils vitest
```

Setup

In each `describe` block where you are taking advantage of the `@nuxt/test-utils` helper methods, you will need to set up the test context before beginning.

```
import { describe, test } from 'vitest'
import { setup, $fetch } from '@nuxt/test-utils'

describe('My test', async () => {
  await setup({
    // test context options
  })
```

```
  test('my test', () => {
    // ...
  })
})
```

Behind the scenes, `setup` performs a number of tasks in `beforeAll`, `beforeEach`, `afterEach` and `afterAll` to set up the Nuxt test environment correctly.

Options

Nuxt Configuration

rootDir

Path to a directory with a Nuxt app to be put under test.

- Type: `string`

- Default: `'.'`

configFile

Name of the configuration file.

- Type: `string`

- Default: `'nuxt.config'`

Setup Timings

setupTimeout

The amount of time (in milliseconds) to allow for `setupTest` to complete its work (which could include building or generating files for a Nuxt application, depending on the options that are passed).

- Type: `number`

- Default: `60000`

Features to Enable

`server`

Whether to launch a server to respond to requests in the test suite.

- Type: `boolean`

- Default: `true`

`port`

If provided, set the launched test server port to the value.

- Type: `number | undefined`

- Default: `undefined`

`build`

Whether to run a separate build step.

- Type: `boolean`

- Default: `true` (`false` if `browser` or `server` is disabled)

`browser`

Under the hood, Nuxt test utils uses `playwright` to carry out browser testing. If this option is set, a browser will be launched and can be controlled in the subsequent test suite. (More info can be found here.)

- Type: `boolean`

- Default: `false`

browserOptions

- Type: `object` with the following properties

  - **type**: The type of browser to launch – either `chromium`, `firefox` or `webkit`

  - **launch**: `object` of options that will be passed to playwright when launching the browser. See full API reference.

runner

Specify the runner for the test suite. Currently, Vitest is recommended.

- Type: `'vitest' | 'jest'`

- Default: `'vitest'`

APIs

APIs for Rendering Testing

$fetch(url)

Get the HTML of a server-rendered page.

```
import { $fetch } from '@nuxt/test-utils'

const html = await $fetch('/')
```

fetch(url)

Get the response of a server-rendered page.

```
import { fetch } from '@nuxt/test-utils'

const res = await fetch('/')
const { body, headers } = res
```

url(path)

Get the full URL for a given page (including the port the test server is running on.)

```
import { url } from '@nuxt/test-utils'

const pageUrl = url('/page')
// 'http://localhost:6840/page'
```

Testing in a Browser

> 🚧 We are working on it, stay tuned!

✎ Edit on Github

Enterprise

Design Kit

NuxtLabs

Nuxt Studio