

Testing Vue with Jest

The guide to happiness

Hello, I am
Roman

@codebryo

@codeship



1

Why should I test?

Why should I test?

- I want to build my application...

Why should I test?

- I want to build my application...
- I test all the time in the **br0w5eR** while developing...

Why should I test?

- I want to build my application...
- I test all the time in the **br0w5eR** while developing...
- Don't have the time...

Why should I test?

- I want to build my application...
- I test all the time in the **br0w5eR** while developing...
- Don't have the time...
- I write perfect code! 🙈🙌

There are many excuses
not to...

... but if something breaks,
you'll learn that customers
don't care for excuses.



Luckily,
testing JS code just got
way easier with JEST.

2

Hello testing...


```
// sum.spec.js
import sum from './sum'

describe('sum', () => {

  it('create sum of 2 numbers', () => {
    expect(sum(15, 8)).toBe(23);
  })

})
```

~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » yarn jest

codebryo@macbryoX

yarn jest v0.19.1

\$ "/Users/codebryo/Projects/Codebryo/presentations/javascript/vueconf/node_modules/.bin/jest"

PASS src/sum.spec.js

sum

✓ creates sum of 2 numbers (3ms)

Test Suites: 1 passed, 1 total

Tests: 1 passed, 1 total

Snapshots: 0 total

Time: 0.778s, estimated 1s

Ran all test suites.

🌟 Done in 1.55s.

~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » █

codebryo@macbryoX

```
// sum.spec.js
import sum from './sum'

describe('sum', () => {

  it('create sum of 2 numbers', () => {
    expect(sum(15, 8)).toBe(23);
    expect(sum('vue', 'conf')).toBe('vueconf')
  })

})
```

```
// sum.spec.js
import sum from './sum'

describe('sum', () => {

  it('create sum of 2 numbers', () => {
    expect(sum(15, 8)).toBe(23);
  })

  it('ignores extra arguments', () => {
    expect(sum(2, 3, 4)).toBe(sum(2, 3))
  })

})
```

~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » yarn jest

codebryo@macbryoX

yarn jest v0.19.1

\$ "/Users/codebryo/Projects/Codebryo/presentations/javascript/vueconf/node_modules/.bin/jest"

PASS src/sum.spec.js

sum

- ✓ create sum of 2 numbers (3ms)
- ✓ ignores extra arguments

Test Suites: 1 passed, 1 total

Tests: 2 passed, 2 total

Snapshots: 0 total

Time: 0.863s, estimated 1s

Ran all test suites.

💎 Done in 1.47s.

~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » █

codebryo@macbryoX



```
// sum.js  
export default function sum(a, b) {  
  return a * b;  
}
```

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » yarn jest
```

codebryo@macbryoX

```
yarn jest v0.19.1
```

```
$ "/Users/codebryo/Projects/Codebryo/presentations/javascript/vueconf/node_modules/.bin/jest"
```

```
FAIL src/sum.spec.js
```

```
  • sum › create sum of 2 numbers
```

```
    expect(received).toBe(expected)
```

```
    Expected value to be (using ===):
```

```
      23
```

```
    Received:
```

```
      120
```

```
    at Object.<anonymous> (src/sum.spec.js:6:39)
```

```
    at Promise.resolve.then.el (node_modules/p-map/index.js:42:16)
```

```
    at process._tickCallback (internal/process/next_tick.js:103:7)
```

```
sum
```

```
  ✕ create sum of 2 numbers (6ms)
```

```
  ✓ ignores extra arguments (1ms)
```

```
Test Suites: 1 failed, 1 total
```

```
Tests:       1 failed, 1 passed, 2 total
```

```
Snapshots:   0 total
```

```
Time:        0.912s, estimated 1s
```

```
Ran all test suites.
```

```
error Command failed with exit code 1.
```

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) »
```

codebryo@macbryoX

Some guardrail questions:

Does the test provide value?

Is the code ready for
edge cases?

What happens if this fails?

3

Testing Vue - Start with Vuex

```
export default {
  state: {
    items: []
  },

  mutations: {
    record(state, item) {
      state.items.push(item)
    },

    completed(state, item) {
      let index = state.items.indexOf(item)
      state.items[index].done = true
    }
  }
}
```



```
import store from './store'

describe('shopping list store', () => {

  describe('mutations', () => {
    test('#record adds passed value to items', () => {
      let mockEntry = {text: 'Milk', done: false}
      let state = { items: [] }

      store.mutations.record(state, mockEntry)

      expect(state.items).toEqual([mockEntry])
    })
  })
})
```

1. Prepare

```
import store from './store'

describe('shopping list store', () => {

  describe('mutations', () => {
    test('#record adds passed value to items', () => {
      let mockEntry = {text: 'Milk', done: false}
      let state = { items: [] }

      store.mutations.record(state, mockEntry)

      expect(state.items).toEqual([mockEntry])
    })
  })
})
```


1. Prepare

2. Run

```
import store from './store'

describe('shopping list store', () => {

  describe('mutations', () => {
    test('#record adds passed value to items', () => {
      let mockEntry = {text: 'Milk', done: false}
      let state = { items: [] }

      store.mutations.record(state, mockEntry)

      expect(state.items).toEqual([mockEntry])
    })
  })
})
```

```
import store from './store'

describe('shopping list store', () => {

  describe('mutations', () => {
    test('#record adds passed value to items', () => {
      let mockEntry = {text: 'Milk', done: false}
      let state = { items: [] }

      store.mutations.record(state, mockEntry)

      expect(state.items).toEqual([mockEntry])
    })
  })
})
```

1. Prepare

2. Run

3. Assert

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » yarn jest src/store.spec.js
```

```
yarn jest v0.19.1
```

```
$ "/Users/codebryo/Projects/Codebryo/presentations/javascript/vueconf/node_modules/.bin/jest" src/store.spec.js
```

```
PASS src/store.spec.js
```

```
  shopping list store
```

```
    mutations
```

```
      ✓ #record adds passed value to items (5ms)
```

```
Test Suites: 1 passed, 1 total
```

```
Tests: 1 passed, 1 total
```

```
Snapshots: 0 total
```

```
Time: 1.447s
```

```
Ran all test suites matching "src/store.spec.js".
```

```
🔦 Done in 2.37s.
```

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) »
```

```
codebryo@macbryoX
```

```
import store from './store'

describe('shopping list store', () => {

  describe('mutations', () => {
    test('#record adds passed value to items', () => {
      ...
    })

    test('#completed updates the `done` attribute', () => {
      let mockEntry = {text: 'Milk', done: false}
      let state = { items: [mockEntry] }

      store.mutations.completed(state, mockEntry)

      expect(state.items[0].done).toBeTruthy()
    })
  })
})
```

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » yarn jest src/store.spec.js
yarn jest v0.19.1
$ "/Users/codebryo/Projects/Codebryo/presentations/javascript/vueconf/node_modules/.bin/jest" src/store.spec.js
PASS src/store.spec.js
  shopping list store
    mutations
      ✓ #record adds passed value to items (4ms)
      ✓ #completed updates the `done` attribute

Test Suites: 1 passed, 1 total
Tests:       2 passed, 2 total
Snapshots:   0 total
Time:        1.459s
Ran all test suites matching "src/store.spec.js".
🌟 Done in 2.33s.

~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) »
```

```
import uuidV4 from 'uuid/v4'

export default {

  state: {
    items: []
  },

  mutations: {
    ...
  },

  actions: {
    record({ commit }, text) {
      commit('record', { text, done: false, uuid: uuidV4() })
    }
  }
}
```




```
import store from './store'
import Vue from 'vue'
import Vuex from 'vuex'

describe('actions', () => {
  let $store

  beforeAll(() => {
    Vue.use(Vuex)
    $store = new Vuex.Store(store)
  })

  describe('#record', () => {
    test('#record should prepare an object', () => {
      $store.dispatch('record', 'Hello')
      console.log(JSON.stringify($store.state.items[0]))
      expect($store.state.items.length).toBe(1)
    })
  })
})
```

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » yarn jest src/store.spec.js
```

codebryo@macbryoX

```
yarn jest v0.19.1
```

```
$ "/Users/codebryo/Projects/Codebryo/presentations/javascript/vueconf/node_modules/.bin/jest" src/store.spec.js
```

```
PASS src/store.spec.js
```

```
  shopping list store
```

```
    actions
```

```
      #record
```

```
        ✓ #record should prepare and object (4ms)
```

```
Test Suites: 1 passed, 1 total
```

```
Tests: 1 passed, 1 total
```

```
Snapshots: 0 total
```

```
Time: 1.051s
```

```
Ran all test suites matching "src/store.spec.js".
```

```
  console.log src/store.spec.js:52
```

```
    {"text":"Hello","done":false,"uuid":"1791099d-1183-4a38-9208-a8861b5fa83f"}
```

```
💎 Done in 1.75s.
```

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » █
```

codebryo@macbryoX

```
{"text": "Hello", "done": false, "uuid": "1791099d-1183-4a38-9208-a8861b5fa83f"}
```

```
jest.mock('uuid/v4')
...
import uuidV4 from 'uuid/v4'

describe('actions', () => {
  let $store

  beforeAll(() => {
    Vue.use(Vuex)
    $store = new Vuex.Store(store)
  })

  describe('#record', () => {
    test('#record should prepare an object', () => {
      uuidV4.mockImplementation(() => 23);

      $store.dispatch('record', 'Hello')

      expect(uuidV4).toHaveBeenCalled()
      expect($store.state.items[0]).toEqual({text: 'Hello', done: false, uuid: 23})
    })
  })
})
```

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » yarn jest src/store.spec.js
```

codebryo@macbryoX

```
yarn jest v0.19.1
```

```
$ "/Users/codebryo/Projects/Codebryo/presentations/javascript/vueconf/node_modules/.bin/jest" src/store.spec.js
```

```
PASS src/store.spec.js
```

```
  shopping list store
```

```
    actions
```

```
      #record
```

```
        ✓ #record should prepare and object (7ms)
```

```
Test Suites: 1 passed, 1 total
```

```
Tests: 1 passed, 1 total
```

```
Snapshots: 0 total
```

```
Time: 0.648s, estimated 1s
```

```
Ran all test suites matching "src/store.spec.js".
```

```
  console.log src/store.spec.js:60
```

```
    {"text":"Hello","done":false,"uuid":23}
```

```
💎 Done in 1.27s.
```

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » █
```

codebryo@macbryoX

```
...
actions: {
  record({ commit }, text) {
    commit('record', { text, done: false, uuid: uuidV4() })
  },

  delayedRecord({ commit }, text) {
    setTimeout(() => {
      commit('record', { text, done: false, uuid: uuidV4() })
    }, 10000)
  }
},
...
```

```
jest.mock(,uuid/v4')
jest.useFakeTimers();
...
import uuidV4 from 'uuid/v4'

describe('actions', () => {
  let $store

  beforeAll(() => {...})

  describe(,#record with delay', () => {
    test('should prepare an object', () => {
      uuidV4.mockImplementation(() => 23);

      $store.dispatch('record', 'Hello')

      jest.runAllTimers();
      expect($store.state.items[0]).toEqual({text: 'Hello', done: false, uuid: 23})
    })
  })
})
```




4

Testing Vue - Components

Where should we start?

- Output is interesting
- Methods should work
- Events should be handled correctly



Snapshot Testing

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » yarn jest src/component.spec.js
```

```
yarn jest v0.19.1
```

```
$ "/Users/codebryo/Projects/Codebryo/presentations/javascript/vueconf/node_modules/.bin/jest" src/component.spec.js
```

```
PASS src/component.spec.js
```

```
✓ snapshot (5ms)
```

Snapshot Summary

```
 › 1 snapshot written in 1 test suite.
```

```
Test Suites: 1 passed, 1 total
```

```
Tests: 1 passed, 1 total
```

```
Snapshots: 1 added, 1 total
```

```
Time: 1.723s
```

```
Ran all test suites matching "src/component.spec.js".
```

```
✦ Done in 2.93s.
```

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) »
```

```
codebryo@macbryoX
```



```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » yarn jest src/component.spec.js
```

codebryo@macbryoX

```
yarn jest v0.19.1
```

```
$ "/Users/codebryo/Projects/Codebryo/presentations/javascript/vueconf/node_modules/.bin/jest" src/component.spec.js
```

```
FAIL src/component.spec.js
```

```
  ● snapshot
```

```
    expect(value).toMatchSnapshot()
```

```
    Received value does not match stored snapshot 1.
```

```
    - "<h1>Hello VueConf</h1>"
```

```
    + "<h1 class='super'>Hello VueConf</h1>"
```

```
      at Object.<anonymous> (src/component.spec.js:29:50)
```

```
      at Promise.resolve.then.el (node_modules/p-map/index.js:42:16)
```

```
      at process._tickCallback (internal/process/next_tick.js:103:7)
```

```
  ✕ snapshot (5ms)
```

Snapshot Summary

```
  › 1 snapshot test failed in 1 test suite. Inspect your code changes or run with `yarn run jest -- -u` to update them.
```

```
Test Suites: 1 failed, 1 total
```

```
Tests: 1 failed, 1 total
```

```
Snapshots: 1 failed, 1 total
```

```
Time: 0.793s, estimated 1s
```

```
Ran all test suites matching "src/component.spec.js".
```

```
error Command failed with exit code 1.
```

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) »
```

codebryo@macbryoX


```
exports[`Shopping-list component snapshot 1`] = `
"<ul class=\\\"shopping-list\\\">
  <li class=\\\"done\\\">Beer<button>Done</button></li>
  <li class=\\\"\\\">Milk<button>Done</button></li>
  <li class=\\\"\\\">Apple<button>Done</button></li>
</ul>"
`;
```

```
exports[`Shopping-list component snapshot 2`] = `
"<ul class=\\\"shopping-list\\\">
  <li class=\\\"done\\\">Beer<button>Done</button></li>
  <li class=\\\"done\\\">Milk<button>Done</button></li>
  <li class=\\\"\\\">Apple<button>Done</button></li>
</ul>"
`;
```

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) » yarn jest src/component.spec.js
```

```
yarn jest v0.19.1
```

```
$ "/Users/codebryo/Projects/Codebryo/presentations/javascript/vueconf/node_modules/.bin/jest" src/component.spec.js
```

```
PASS src/component.spec.js
```

```
  Shopping-list component
```

```
    ✓ snapshot (31ms)
```

```
    ✓ click (13ms)
```

```
Test Suites: 1 passed, 1 total
```

```
Tests: 2 passed, 2 total
```

```
Snapshots: 2 passed, 2 total
```

```
Time: 0.72s, estimated 1s
```

```
Ran all test suites matching "src/component.spec.js".
```

```
🌟 Done in 1.43s.
```

```
~/Projects/Codebryo/presentations/javascript/vueconf(branch:master*) »
```

```
codebryo@macbryoX
```

We learned

- How to structure Jest specs
- How to test Vuex stores
- How to mock dependencies
- How to snapshot rendered output
- How to travers the DOM and trigger events

BOARDS BOARD

```
const Component = {  
  props: ['foo'],  
  
  render(h) {  
    return h('div', this.foo)  
  }  
}
```



```
let C = Vue.extend(Component)

$mounted = new C({
  propsData: {foo: 'bar'}
}).$mount()
```

```
const StoreVue = Vue.extend()  
StoreVue.use(Vuex)
```

```
Component.store = new Vuex.Store(store)
```

```
const $mounted = new StoreVue(Component).$mount()
```

Just the beginning!

vue-test-utils

<https://github.com/vuejs/vue-test-utils>



Thank you

@codebryo | @codeship