JS Unit Test Setup in a Ruby on Rails Application

Abhinav

Growth Session #13 - March 15-16 2018

- Add Javascript unit test setup inside a **Ruby on Rails App**

- Add Javascript unit tests for base JS components in **Vcliq**

Achievements and progress: JS Test Tools

- [Karma](#) - Test Runner for JavaScript.

  ⇒ Supported Browsers: **Chrome** (for local development) & **Chrome Headless** (for CI)

- [Webpack](#) - Module Bundler for JavaScript modules

- [Jasmine](#) - JavaScript Test Framework

- [Jasmine-dom-matchers](#): Matching library for making assertions about HTML in Jasmine tests

- [Istanbul](#): JS code coverage tool

```
frameworks: ['jasmine', 'jasmine-dom-matchers'],
plugins: [
  'karma-jasmine',
  'karma-jasmine-dom-matchers',
  'karma-webpack',
  'karma-chrome-launcher',
  'karma-sourcemap-loader',
  'karma-spec-reporter',
  'karma-coverage-istanbul-reporter'
],
  // list of files / patterns to load in the browser
  files: [
    'app/assets/javascripts/polyfills/htmlCollectionIterator.js',
    'app/assets/javascripts/polyfills/nodeListIterator.js',
    'spec/support/jasmine_plugins/**/*.js',
    'spec/javascripts/**/*_spec.js'
  ],
```

Test Setup - Core Karma Configurations

```
browsers: ['Chrome', 'ChromeHeadlessNoSandbox'],

    customLaunchers: {
      ChromeHeadlessNoSandbox: {
        base: 'ChromeHeadless',
          flags: ['--no-sandbox']
      }
    },

    reporters: ['spec', 'coverage-istanbul'],
```

Sandbox Jasmine Plugin

```javascript
window.JasmineSandbox = window.JasmineSandbox || {
  containerId: 'jasmineSandbox',

  mount: function(htmlString) {
    let sandboxElement = document.querySelector(`#${JasmineSandbox.containerId}`);

    if (!sandboxElement) {
      sandboxElement = document.createElement('div');
      sandboxElement.setAttribute('id', JasmineSandbox.containerId);
      document.body.appendChild(sandboxElement);
    }

    sandboxElement.innerHTML = htmlString;
  }
};

/* eslint-env jasmine */
afterEach(function() {
  let container = document.querySelector(`#${JasmineSandbox.containerId}`);

  container && container.parentNode.removeChild(container);
});
```

Test Scripts

- Tests are run using **npm scripts**.

- Following scripts are available:

```
"test": "NODE_ENV=test karma start --browsers ChromeHeadlessNoSandbox"

"test:debug": "NODE_ENV=test karma start --browsers Chrome --singleRun=false"
```

Spec Files Organization

```
spec/javascripts
├── backButton
│   ├── index_spec.js
│   └── template.js
├── dropdown
│   ├── index_spec.js
│   └── template.js
├── modal
│   ├── index_spec.js
│   └── template.js
├── navTab
│   ├── index_spec.js
│   └── template.js
├── progressiveImageLoader
│   ├── index_spec.js
│   └── template.js
└── selectInput
    ├── index_spec.js
    └── template.js
```

Spec Samples - DOM Matcher

```javascript
import Modal from 'CoreComponents/modal/';
import Template from './template';

it('creates an overlay', () => {
  JasmineSandbox.mount(Template['default']);

  new Modal(document.querySelector('.modal'));

  expect('.modal__overlay').toExist();
});
```

Spec Samples - Event Triggers

```javascript
import Modal from 'CoreComponents/modal/';
import Template from './template';

describe('#show', function() {
    it('adds SHOW classname to the modal element', () => {
        JasmineSandbox.mount(Template['default']);
        let modal = new Modal(document.querySelector('.modal'));

        modal.show(new MouseEvent('click'));

        expect(modal.elementRef).toHaveClass('modal--show');
    });
});
```

Spec Samples - Spies

```javascript
import Modal from 'CoreComponents/modal/';
import Template from './template';

describe('when modal is dismissible', function() {
  it('invokes modal hide', () => {
    JasmineSandbox.mount(Template['default']);
    spyOn(Modal.prototype, 'hide');

    let modal = new Modal(document.querySelector('.modal'));

    modal.overlayClickHandler(new MouseEvent('click'));

    expect(Modal.prototype.hide).toHaveBeenCalled();
  });
});
```

Spec Execution - CLI

```
    Dropdown
      #hide
        ✓ hides the dropdown
      #toggle
        when dropdown is open
          ✓ closes the dropdown
        when dropdown is NOT open
          ✓ opens the dropdown
        ✓ updates the popper instance of the dropdown
      Initialization
        ✓ adds toggle event listener to the dropdown element
        ✓ adds hide event listener to body
        ✓ assigns instance variables

    SelectInput
      Initialization
        select--styled
          when NO option is selected
            ✓ adds placeholder class to the wrapper
            ✓ adds placeholder text

HeadlessChrome 65.0.3325 (Mac OS X 10.13.3): Executed 44 of 44 SUCCESS (0.131 secs / 0 secs)
TOTAL: 44 SUCCESS


============================= Coverage summary ===============================
Statements   : 88.15% ( 186/211 )
Branches     : 84% ( 42/50 )
Functions    : 87.5% ( 49/56 )
Lines        : 88.94% ( 185/208 )
==============================================================================
```
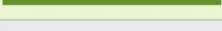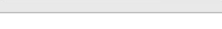
Spec Execution - Coverage Generated by Istanbul

All files

88.15% Statements `186/211`   **84%** Branches `42/50`   **87.5%** Functions `49/56`   **88.94%** Lines `185/208`

Press *n* or *j* to go to the next uncovered block, *b*, *p* or *k* for the previous block.

File ▲		Statements ⇕		Branches ⇕		Functions ⇕		Lines ⇕	
backButton		100%	8/8	100%	0/0	100%	4/4	100%	8/8
dropdown		100%	21/21	60%	3/5	100%	5/5	100%	21/21
modal		100%	39/39	93.33%	14/15	100%	12/12	100%	38/38
navTab		100%	31/31	100%	7/7	100%	12/12	100%	31/31
progressiveImageLoader		100%	16/16	100%	0/0	100%	6/6	100%	16/16
selectInput		73.96%	71/96	78.26%	18/23	58.82%	10/17	75.53%	71/94

Thanks!

Contact Nimbl3

hello@nimbl3.com

399 Sukhumvit Road, Interchange 21
Klongtoey nua, Wattana
Bangkok 10110

20th Floor, Central Tower
28 Queen's Road
Central, Hong Kong

nimbl3.com