

Testing Vue Components with Jest

Setting up a Jest Testing Suite for Vue Applications



Daniel Stern

The Code Whisperer

<http://danielstern.ca/social-media>

A Vue-Jest Testing Scenario



Globoticket sells concert tickets online

Globoticket's website started off simple, but is now complex and fast expanding

New feature, a complicated forum, will be added

Project manager has been manually testing for errors and regression

- Process used to take 5 minutes, now takes 2 hours**

A Vue-Jest Testing Scenario



The CEO has asked you to save the valuable time of QA and the Project Manager

- Doing so, you will also make the testing process more effective at preventing errors**

The solution is to add Jest to this Vue application

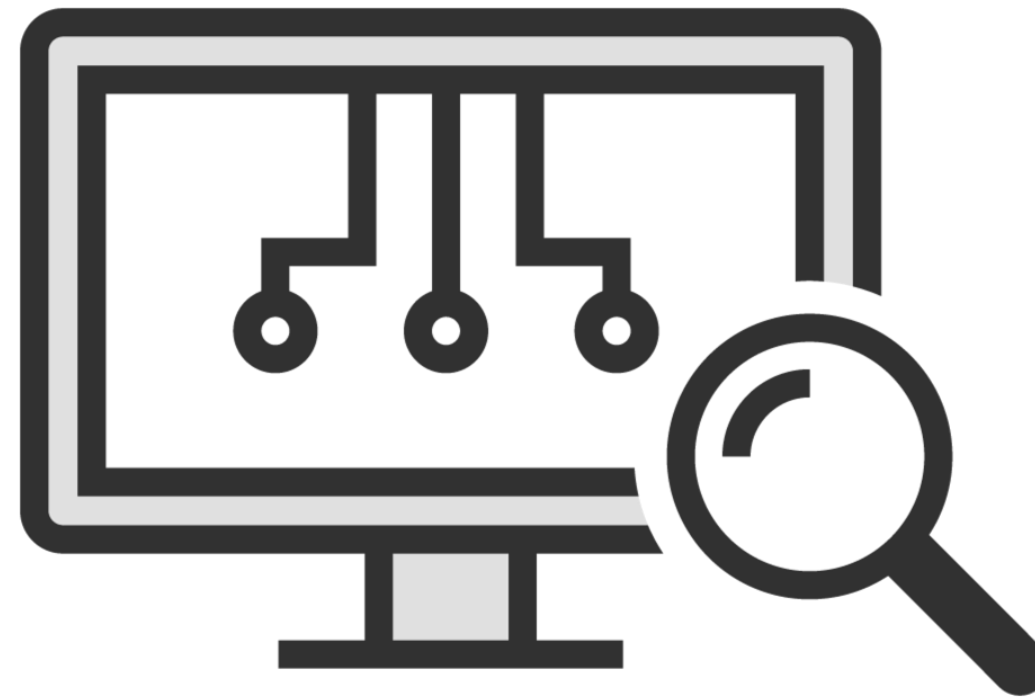
Properly configured, Jest will automatically verify the application works before each commit and push to production

Course Roadmap



1.

**Install Jest and set up
a functional test suite
for a Vue application**



2.

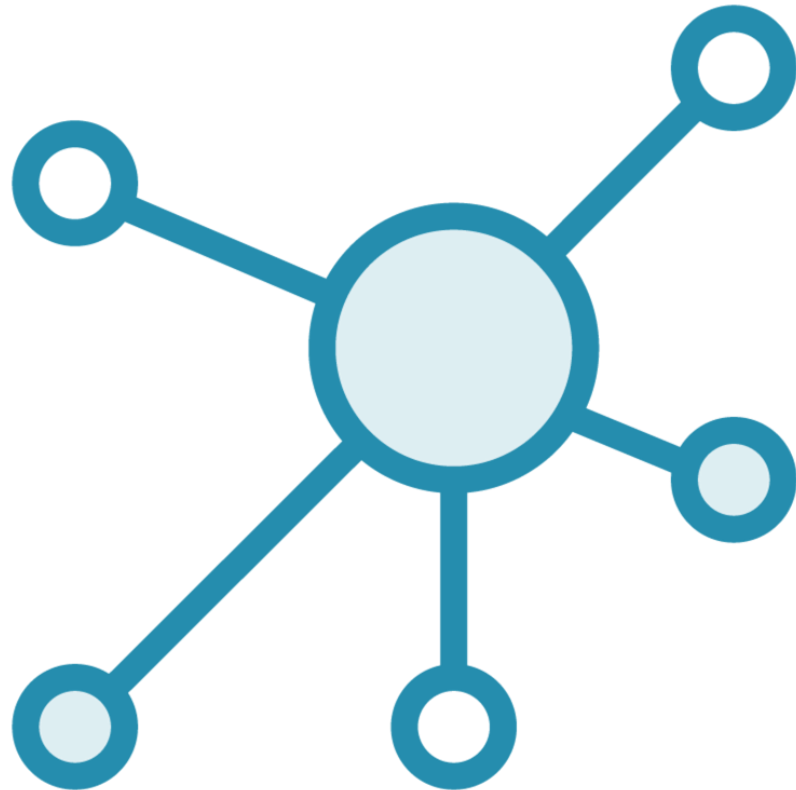
**Use Jest to test
components and
services within Vue
application**



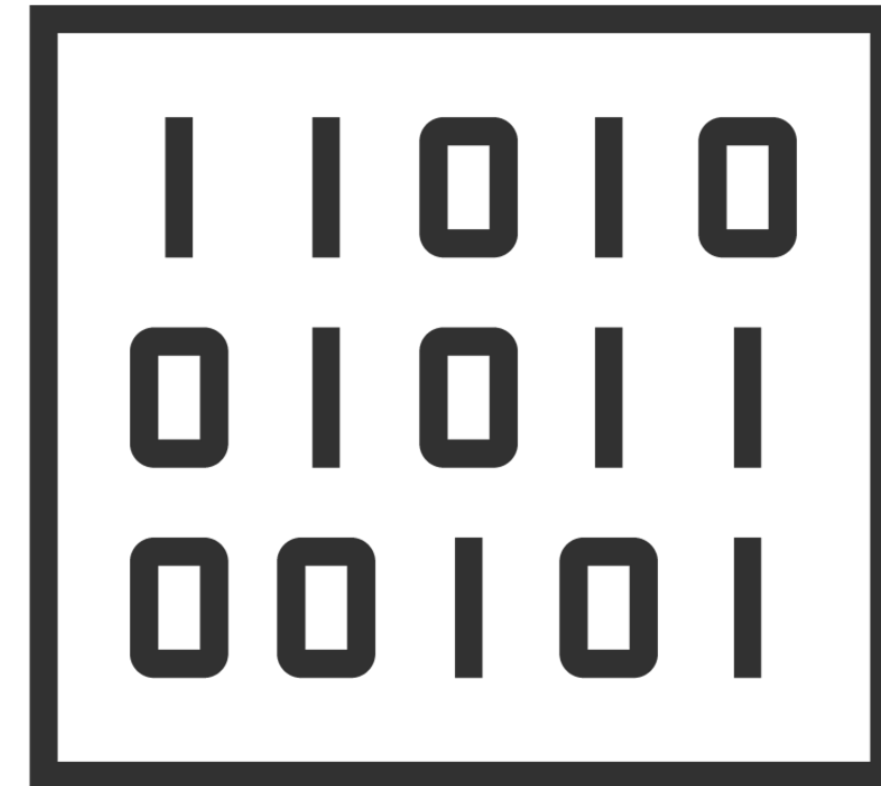
3.

**Integrate
application with
Continuous
Integration**

Learning Objectives



Install, configure and integrate Jest with a boilerplate Vue / Node.js project

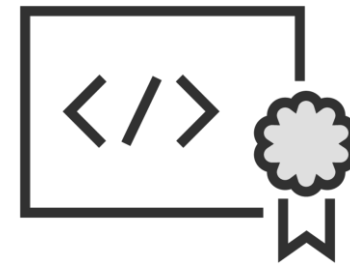


Understand setup / teardown tools (*afterEach*, etc.) and use them to create functional test stubs

Before Starting, You Should Be Familiar With...

It is recommended you understand these other tools, as they are foundational in our understanding of testing Vue applications with Jest.

General knowledge of Jest is an asset but not required – all necessary code will be explained.



**Intermediate JavaScript
(closures, async, iteration, etc.)**



Vue, a JavaScript MVC library



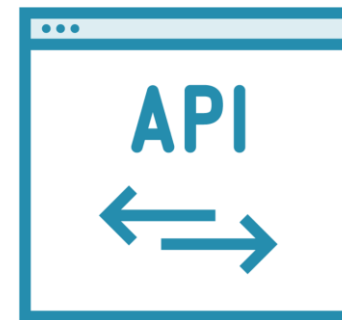
**Node.js, a JavaScript runtime
used to serve applications
and countless other things**

Installing and Running the Demo Application

Understanding the Project Files

The project files are designed to be simple to install and understand.

They have been designed to omit all unnecessary details, so every line of code is worth understanding.



Part of a Vue Application consisting of an interactive messenger service



Initially contains no tests (we will add tests throughout course)



Node.js Application – remember to install Node, dependencies

Options for Downloading the Project Files

1

Download reference materials via link section for this course

2

**Download from primary backup at
<https://github.com/danielstern/vue-jest>**

or 3

**Download secondary backup from
<http://danielstern.ca/vue-jest>**

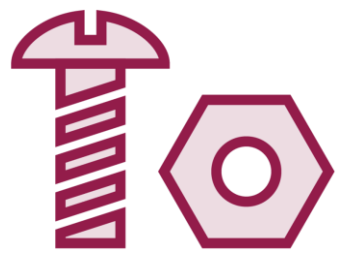
Troubleshooting the Project



Node – latest version or v12.18.3 compatible



Globally installed CLI tools (i.e., jest-cli) can create baffling conflicts with locally installed tools with the same name
Solution: Use npm scripts, which ignore global installs



Vue / Jest project necessarily contains many moving parts (Jest, Babel, Vue, Webpack) which are not trivial to troubleshoot
Solution: Troubleshoot exceptions as they arise



Share problems and solutions at this project's GitHub repository:
<https://github.com/danielstern/vue-jest>

Demo



Download project folder from GitHub

Switch to demo branch with no tests

Install dependencies

Troubleshoot any malfunctioning tools

Explore folder structure and files

Preview application using `npm start`

Installing and Configuring Jest



Brief Review Jest and Tests

We have a huge amount of Vue-Jest specific data to cover in a short timeframe, so we will review these high-level concepts only briefly.

Review: What Is Jest?



Modern test runner and framework (replaces Mocha, jUnit, etc.)



Basic usage: Invoke Jest in Terminal – process exits with error if any tests fail

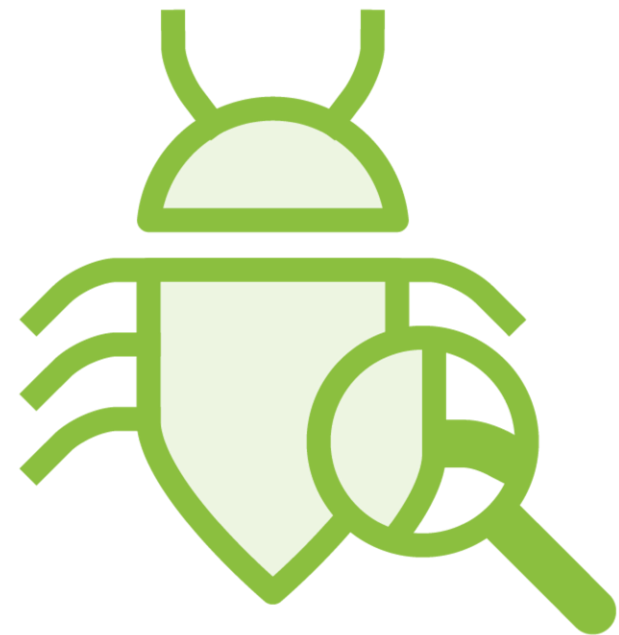


Standard and advanced testing features (mocking, snapshots, etc.)

You probably have a basic understanding of what Jest is if you have elected to take this course.

Let's quickly review and refresh our understanding.

Review: What Are Tests?



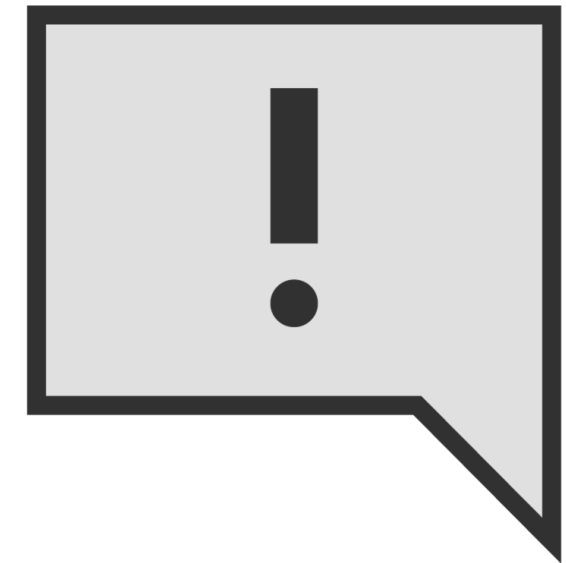
A test is any
method
containing
assertions

$$\cancel{2} + \cancel{2} = \cancel{5}$$

Assertions
throw an error
if two values
don't match*



Fast, instantly
checking
thousands of
assertions



Assertions fail
where code
or tests are
broken

Installing and Configuring Jest



Install Jest using NPM



Update package.json **with** npm test **and add** jest.config.js



Create test stubs for planned test suites



Install project-level and IDE-level tools for Jest development

Demo



Install Jest using NPM

Install Jest typings

**Install any desired tools for IDE
(i.e. , vscode-jest)**

Add npm script for testing

**Add jest.config.js to manage project-
specific configuration**

**Create empty test “stub” for
chat component**

Setting up a Test Suite

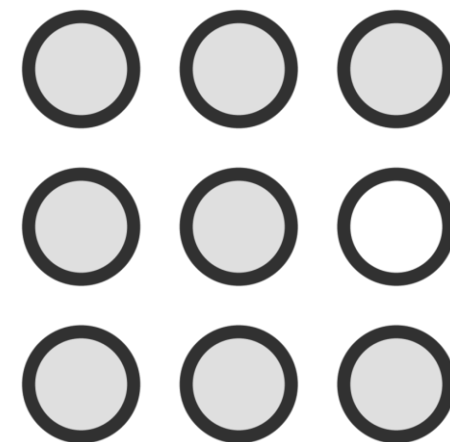
What Are Test Suites?

Test Suites are tools for grouping together tests. Different test suites can have different conditions for running or can have different consequences in event of a test failing.



Focused

Each suite tests a particular component or service



Independent

Suite can run on its own, allowing developer to focus on affected areas



Descriptive

Named suites allow developers, QA to identify which area of site a failing test relates to

Demo



Create coverage consisting of 1 test suite (via describe) containing 1 test

- **Spec files which do not contain any suites and suites which do not contain any tests will both cause an error**

Verify our test suite is invoked when running global Jest script

Summary



Automated unit testing is needed whenever the burden of human testing becomes too great

Jest is a test runner which fails with an error code if any tests fail

Jest can be installed in any NPM-based Vue application where tests are needed

npm test script is well-suited for invoking Jest

Jest requires project-specific configuration to work with Vue

Suites contain a number of tests, which in turn contain a number of assertions