Jest và quy trình làm việc với Jest

1. Jest là gì?

* Jest là một thư viện dùng cho việc testing dự án dùng Javascript. Được tạo ra bởi facebook. Được sử dụng trong các dự án có sử dụng: Babel, TypeScript, Node, React, Angular, Vue, …

1. Ưu và nhược điểm
2. Ưu điểm

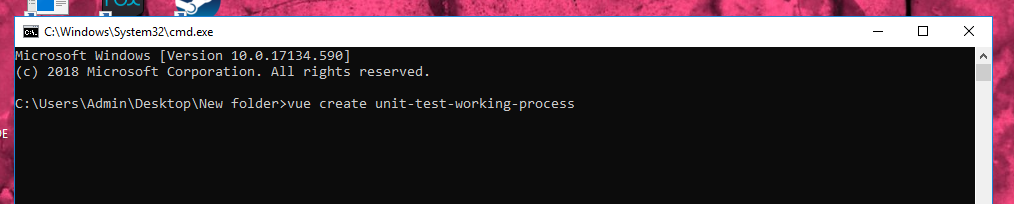
* Không cần cấu hình
* Snapshots: thực hiện các theo dõi đối tượng lớn dễ dàng.
* Cô lập trong quá trình xử lý: các bài test được chạy song song để tối đa hiệu suất.
* All in one: một mình Jest là đã có đầy đủ test runner, assert và mock. Ngoài ra còn có thêm cả Coverage reports…
* Nhanh

1. Nhược điểm

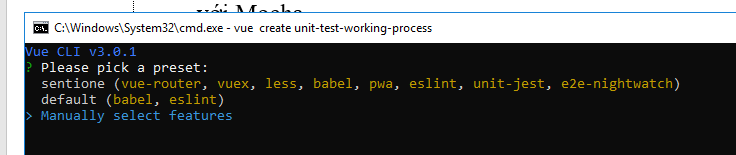
* Mới, ít tool, library và chưa được dùng nhiều bởi cộng đồng developer so với Mocha

1. Cài đặt và sử dụng
2. Tạo project có Jest trong Vue-Cli

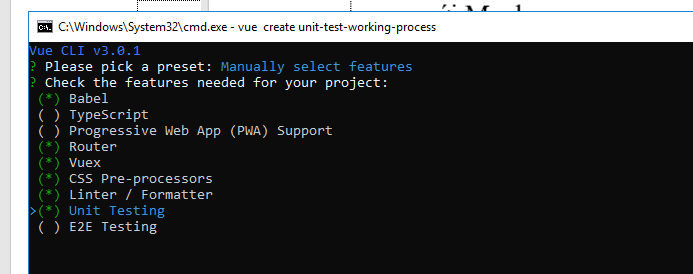
* Mở cmd và thực hiện dòng lệnh sau:



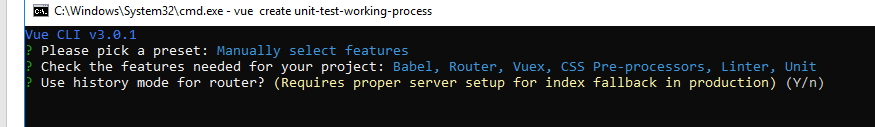
* Chọn manually select features sau đó Enter



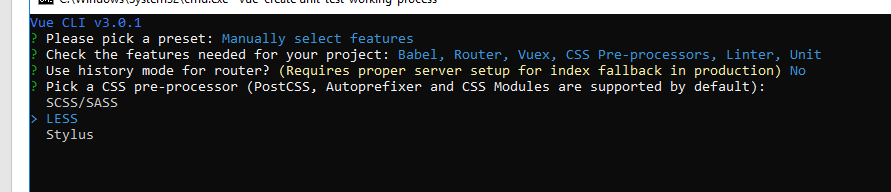
* Chọn unit testing và các tính năng khác nếu cần bằng phím cách



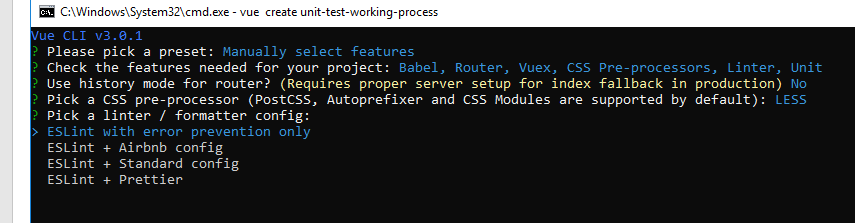
* Chọn Yes/No



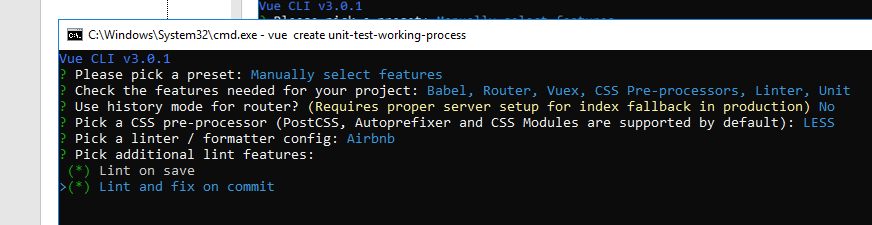
* Chọn Less



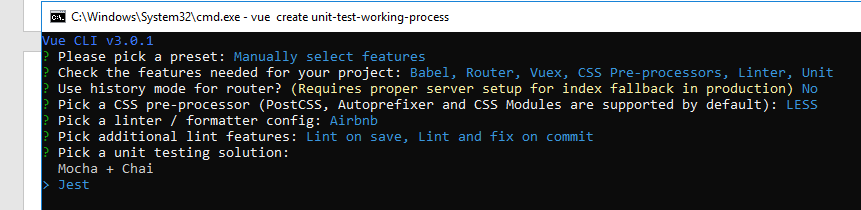
* Chọn 1 Eslint bất kì



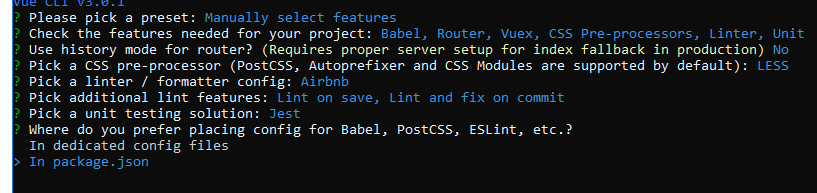
* Chọn 1 trong 2 hoặc cả 2 đều được



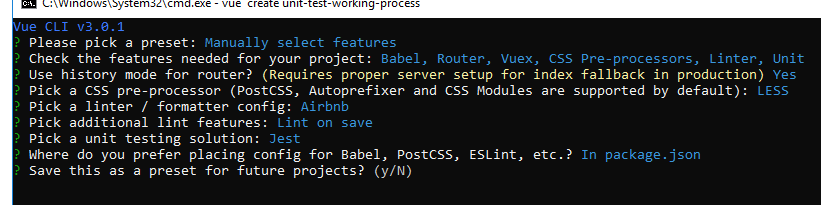
* Chọn Jest



* Chọn package.json



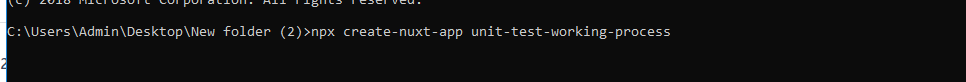
* Chọn Yes/No



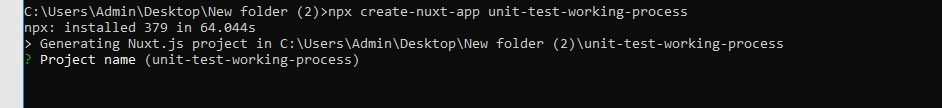
Đợi cho quá trình hoàn tất.

1. Tạo project có Jest trong Nuxt

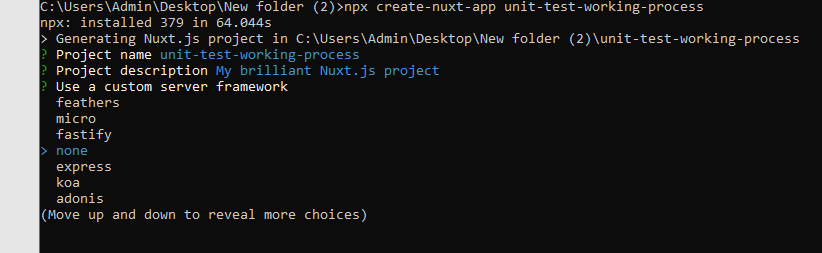
* Vào cmd và thực hiện dòng lệnh sau:



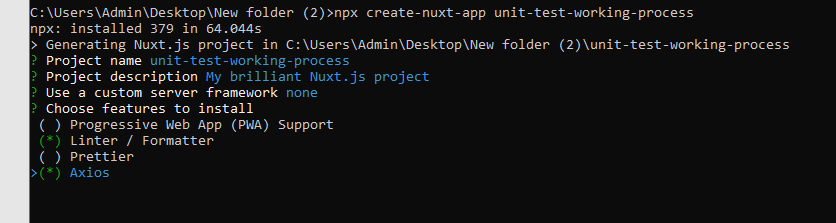
* Đợi 1 chút. Thấy hiện như hình dưới thì Enter 2 lần



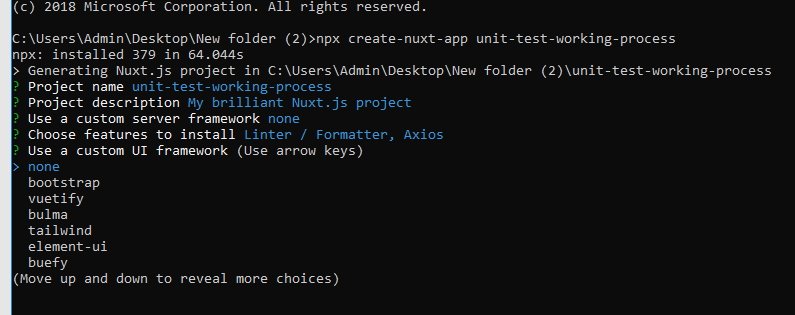
* Chọn một server framework. ở đây ta chọn none



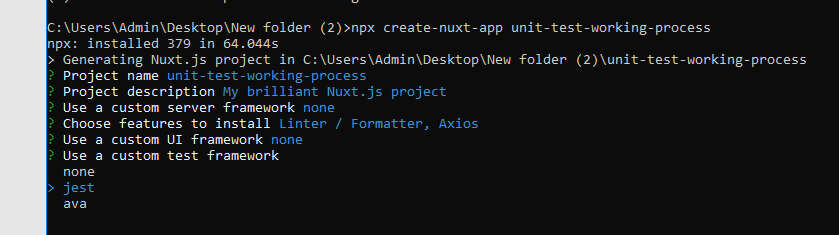
* Chọn nhưng tuỳ chọn cần thiết.



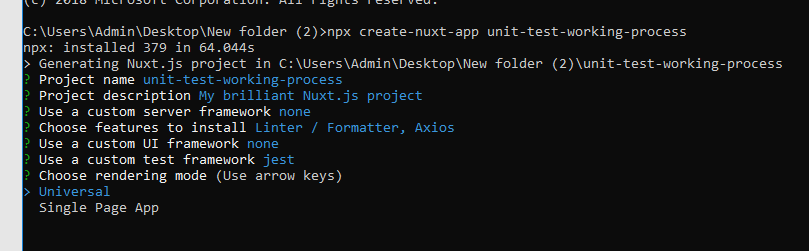
* Chọn những UI framework. Ở đây chọn none



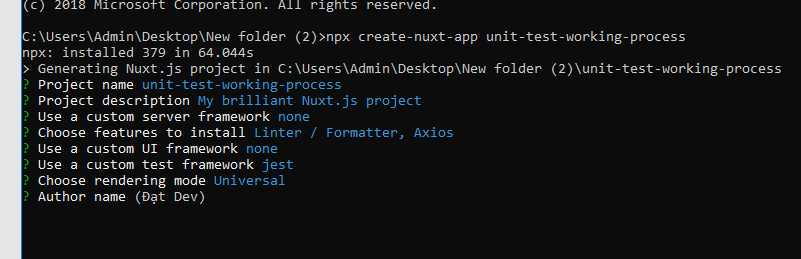
* Chọn Jest



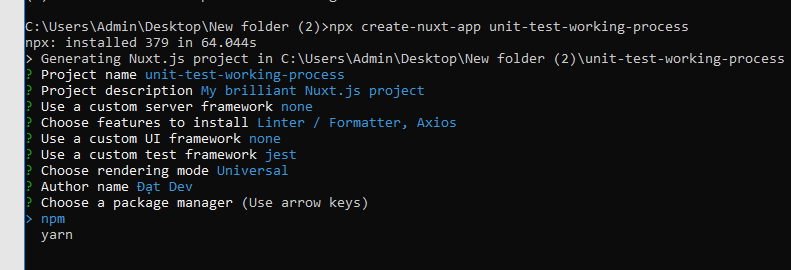
* Chọn 1 trong 2 Mode



* Enter khi thấy hình dưới:



* Chọn package manager. Ở đây ta chọn NPM



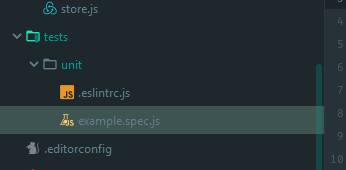
* Đợi cho việc cài đặt hoàn tất.

1. Cách sử dụng và viết Unit test

Sau khi ta viết 1 component xong. Ta cần viết 1 file unit test cho riêng component này. Viết unit test được dựa theo cách test case, kịch bản có trước khi ta code để kiểm tra component ta vừa code xong có đúng với yêu cầu đã được bàn bạc trước.

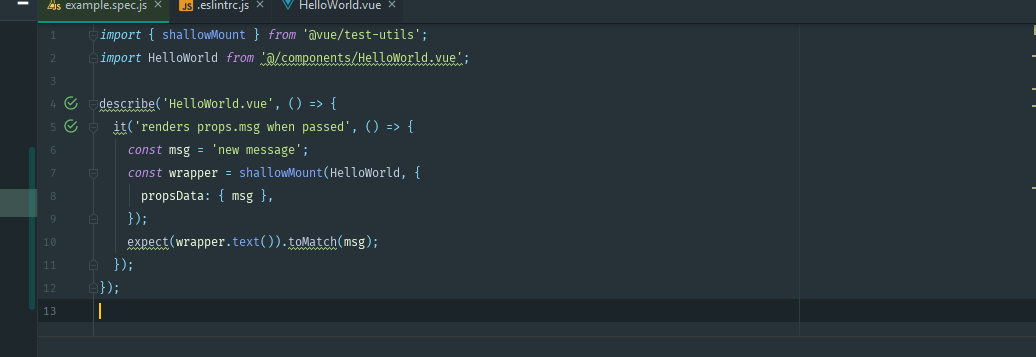
Để viết test, ta vào trong thư mục test và tạo ra file có đuôi là .spec.js

VD:



Và viết các dòng code

VD:



Ở trên, ta đang viết code unit test cho trang Hello World

Mô tả:

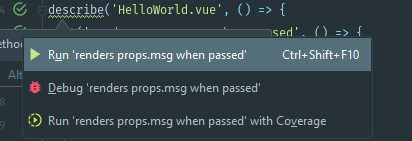
* describe: Mô tả cho component đang được test
* một test case được viết trong hàm it và trả về message(renders props.msg when passed) nếu test case không xảy ra lỗi nào.
* Biến wrapper là kết quả trả về của hàm shallowMount(hàm dùng để render ra code của component ta vừa viết).
* Bên trong ta có expect(…).toMatch(…) là cú pháp viết để chỉ ra code trong component có match với kết quả mà ta biết trước hay không. Tham số trong expect là code component được render ra. Tham số trong toMatch là kết quả component được vẽ ra trước.

Trong một describe ta có thể viết được nhiều test case it trong đó.

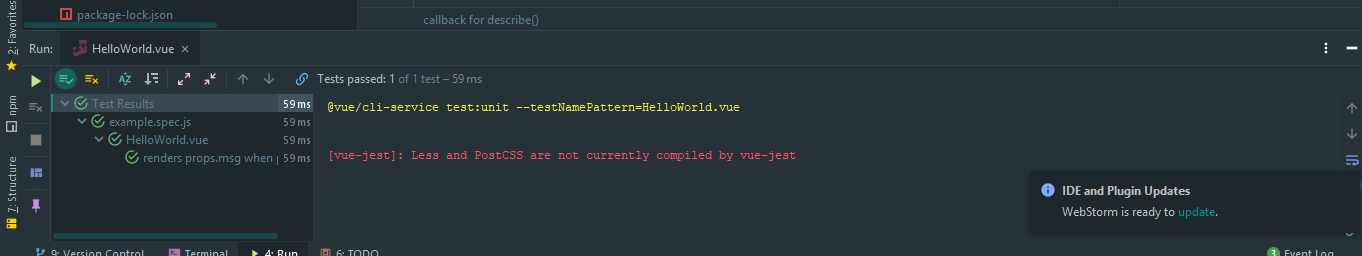
Để chạy một test case ta click vào hình ở bên trái test case như trong hình



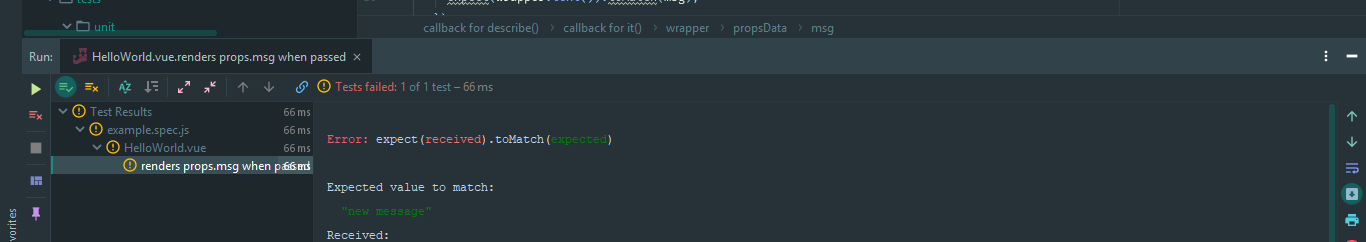
Sau đó chọn Run như hình



Kết quả của test case sẽ hiện như hình dưới. Ở đây test case của ta đúng:



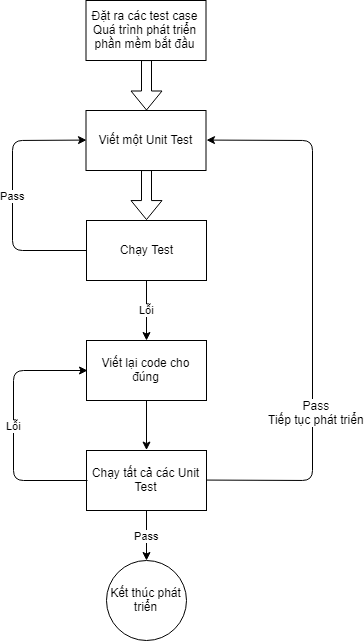
Ngược lại nếu ta thêm vào toMatch một kết quả khác để xem lỗi:



Tài liệu về cách viết unit test trong Vue chúng ta có thể tham khảo:

<https://vue-test-utils.vuejs.org/>

1. Sơ đồ quy trình làm việc với Jest Unit Test



1. Demo