NodeJS

EMERGING TECHNOLOGIES

(TECH8020)
(Embedded Systems Development)
(Fall-2021)

ASSIGNMENT - 4

SUBMITTED BY:

Sai Teja Anaparthy (8696855)

NodeJS library - JEST

INTRODUCTION:

Jest is a test runner library, and this is used for creating the tests and running the test and for the structuring the tests. The jest is used mostly now-a-days everywhere and its best choice for react users and it doesn't depend upon on the any third - party applications for its proper working. It doesn't have any configuration for its setup we can simply start writing the test code after its installation.

CHARACTERISTICS:

- Jest has zero configuration for setting it up and its works on most of the JavaScript projects and we just need to install zest as dependency of the project.
- In zest the tests are isolated from each other because they are executed parallelly and don't influence on the other results and it's also working a collector which collects the tests results from all the test processes.
- Jest also have the snapshot testing which is to verify the large objects correctness by this feature it makes one the most powerful feature.
- Jest also offers a rich APIs for various specific needs by the user.

INSTALLATION:

First, we need to install node in the system and if you don't have it here is the link for Node Js

https://docs.npmjs.com/downloading-and-installing-node-js-and-npm

With help of npm we can easily download the jest

npm install –-save-dev jest

To run from the command line, first we need to use this command

npm install jest –-global

To run jest on files

Jest my-test –-notify --config=config.json

We need to generate basic generation file

Jest --init

To know the jest version

Jest -version

OPERATION:

We will write a basic test code for usage of the jest library.

We need to install globally first:

npm install -g jest

We will create a folder and we will open it:

Mkdir testing-jest. (I named testing you can name anything you like)
Cd testing-jest

We will create the project:

npm init -y

First, we will create a basic file **subtract.js** which will subtract two values:

```
function subtract (c, d) {
  return c + d;
}
module.exports = subtract;
```

We will add the jest to project as dependency:

npm install –save-dev jest

We will create another file which contains the testing code, and we need the file name as **subtract.test.js**:

```
const subtract = require('./subtract');

test ('subtracts 3 - 4 to equal 1', () => {
```

```
    expect (subtract (3, 4)).toBe(1);
});
```

Now we need to add jest to json package

```
{
  "scripts": {
    "test": "jest"
  }
}
```

Then we need to run the test by **npm run test** command in command line

```
PASS  ./subtract.test.js
✓ subtracts 3 - 4 to equal 1 (5ms)
```

This is the similar kind of output you expected see on the screen after running the command on the command line.

ADVANTAGES:

- It offers a command line interface to control your tests easily.
- It also contains the interactive mode which will all the tests for the recent code you have returned.
- It also has syntax for testing the single test of the code or we can just simply skip the test by using syntax like **.skip**

CONCLUSION:

Jest is the complete testing framework, and it supports various style of testings, and it consists of very key features like snapshot which makes it as one of the best testing tools for JavaScript.

REFERENCE:

[N.N, (August 27,2021). Jest Testing: A Helpful, Introductory Tutorial. Retrieved from https://www.testim.io/blog/jest-testing-a-helpful-introductory-tutorial/]

[Valentino Gagliardi, (July 25,2020). Jest Tutorial for Beginners: Getting Started with JavaScript Testing. Retrieved from https://www.valentinog.com/blog/jest/]

[N.N, (2021). Jest: Getting Started. Retrieved from https://jestjs.io/docs/getting-started]