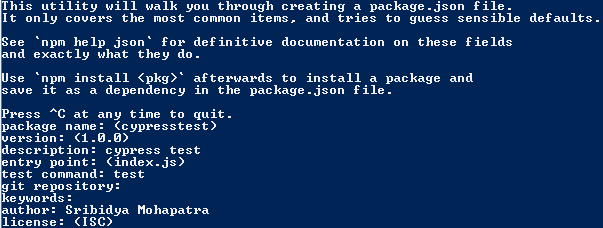
**Table of Contents**

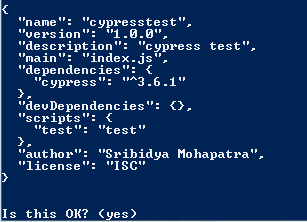
1. Installation of Cypress 2
2. Writing the Test Script 4
3. Running the test Script 4
4. TestCase\_01 : Ease-Live 5
5. TestCase\_02 : Contact 6
6. Like and Dislike About Cypress 6

**1. Installation of Cypress**

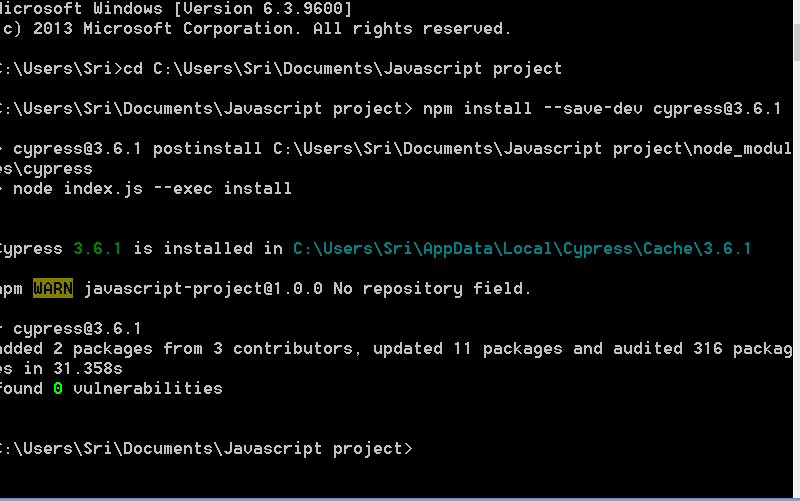
Cypress is a desktop application installed in the computer. Installation of Cypress can be done via npm.

1. To install Cypress via npm we have to first install Node.js
2. <https://nodejs.org/en/download/> gives the latest version of Node.js.
3. The version for Node.js is 12.13.1 (includes npm 6.12.1)
4. After installation of Node.js we create a folder to install Cypress.
5. In windows we open the PowerShell or command prompt and go to the folder path.
6. Before we execute the installation command we run first the npm init command or have a node\_modules folder or package.json file in the root of our project as it ensures we have installed Cypress in the correct directory.
7. We type **npm init** in the command prompt and it lets us to create package.json file.





1. After typing yes it creates the package.json file.
2. Next we install Cypress. The version of the Cypress is 3.6.1
3. We type **npm install cypress –save—dev** in command prompt. This will install cypress with dev dependency needed for the project and update the package.json file.

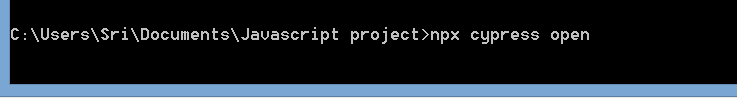


1. After successful installation we open the Cypress application.

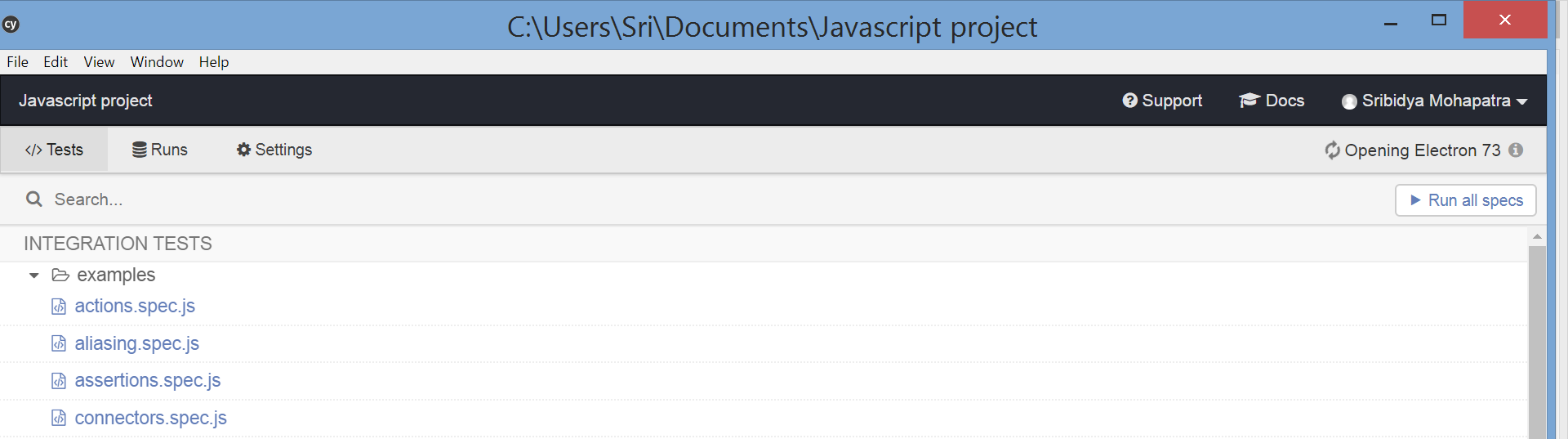
**For this test I have updated package.json as below**

***scripts": {***

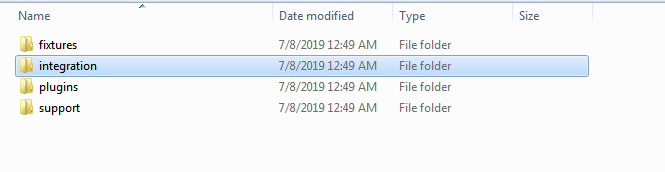
***"cypress:open": "cypress open",***



1. When the application options it looks like the below image:



1. When we open the folder in the drive where we have installed cypress the folder structure looks like the below image:



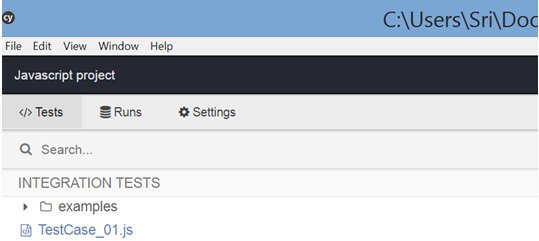
1. The test scripts that are written and run will be in the integration folder.

**2. Writing the Test Script**

1. To write the test scripts we can use any text editor like Sublime Text, Visual Studio Code etc.
2. In this project Sublime Text version 3.2.1 is used as text editor.
3. After we open the editor we have to navigate to the Cypress installation folder and then to the integration folder.
4. We create JavaScript file with name TestCase\_01.js and the code for the test script is written in it.

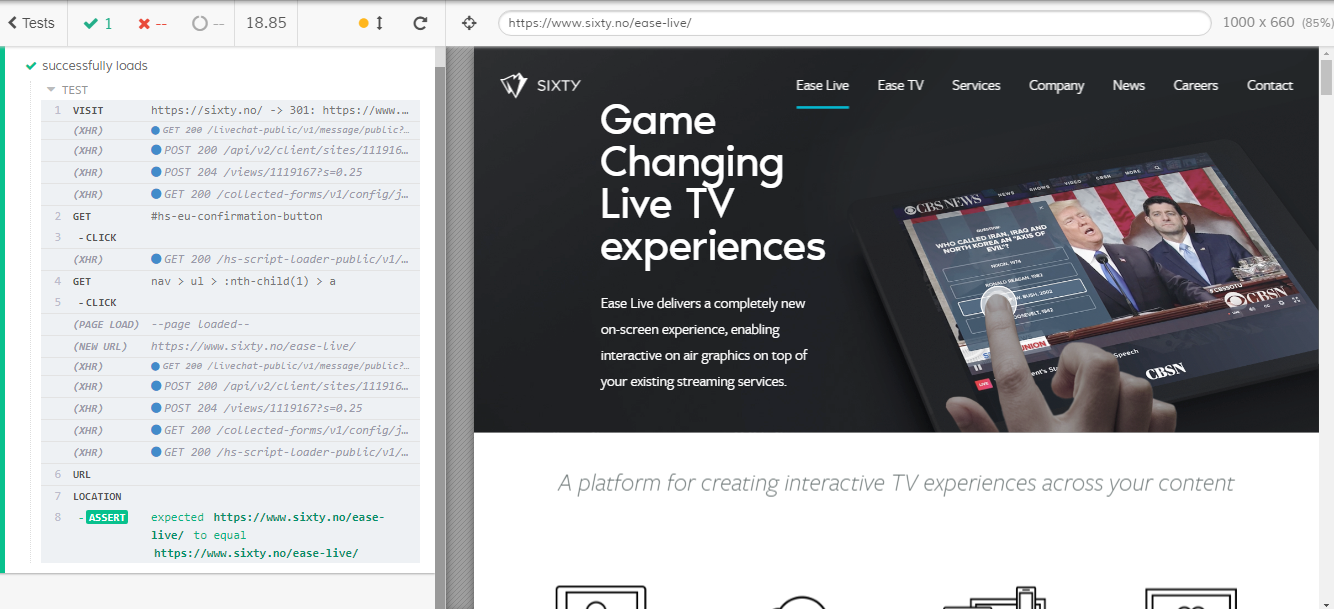
**3. Running the Test Script**

1. After the test script is written in text editor we go to the Cypress app to test it.
2. In Cypress app we see the file with name TestCase\_01.js and to run it we click the file.



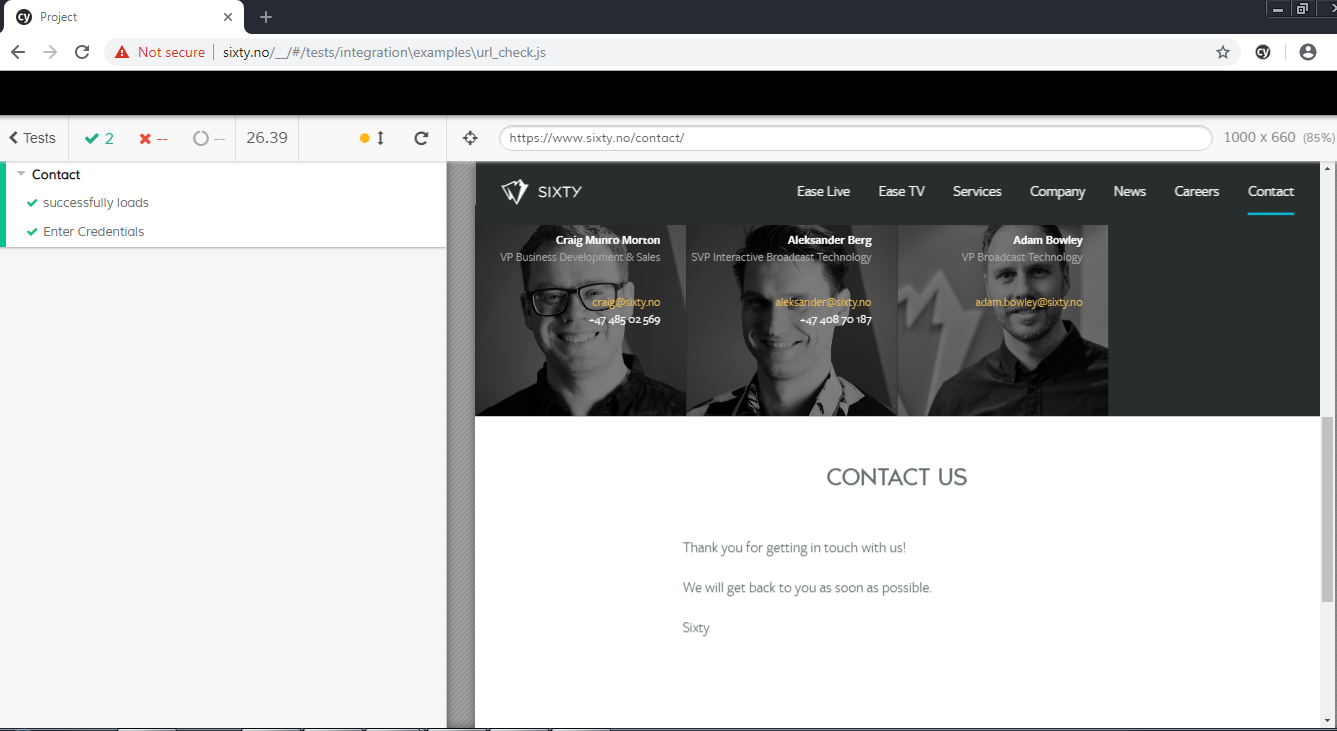
1. Cypress will automatically run the scenario or the test script in the browser directly.
2. **TestCase\_01: Ease-Live**

Opens sixty.no, clicks on Ease Live and verifies the url of the Ease Live page should be <https://www.sixty.no/ease-live/> . Below image displays the successful execution of the test case.



**5. TestCase\_02: Contact**

Opens sixty.no, clicks on Contact and tests successful submission of contact form by filling out the required details in the form. Below image displays the successful execution of the test case.



**6. Like and Dislike About Cypress**

1. Like about Cypress:

* Setting up Cypress is very easy and straightforward. Install Node.js, define package.json file as a dependency and we start using Cypress.
* Step by step execution of scripts which makes it easier to debug events and provide a detailed report.
* Cypress is open source.
* Its fast as it uses in-browser execution where you have just one process: the browser itself that runs alongside the App code.

2. Dislike about Cypress:

* Cypress tests can only be written in JavaScript. No other languages are supported.
* Mocha is the testing framework that you write your tests with .You cannot write your tests in another JavaScript framework, such as Jest or Tape.
* The Cypress test runner only works on Chrome. It has no support for Firefox, Safari, Edge, or IE yet