1. Install Dependencies:
2. Install Node.js and npm
3. Verify the nodejs path is added to the environment variables
4. Check the node and npm version using the commands:

node -v

npm -v

1. Clone the repository to your machine using the below command:

git clone -b master <url>

1. Move to the project directory cloned from the git and run the below command to install the dependencies :

npm install

1. Install and Configure Cypress:
2. Change the directory to the project directory
3. Install Cypress as a development dependency

npm install cypress --save-dev

1. Open the Cypress using the command

npx cypress open

1. Configure End to End Testing or Component Testing as required(details are mentioned in the later sections)
2. This configuration will create Cypress folder and cypress.config.json file
3. Creation of E2E Test Cases:
4. Open the Cypress using the command

npx cypress open

1. Select End to End Testing option
2. Select any browser as required
3. Create a spec under Cypress/e2e folder
4. Write the required test script for one basic end to end flow of the application
5. Creation of Component Test Cases:
6. Open the Cypress using the command

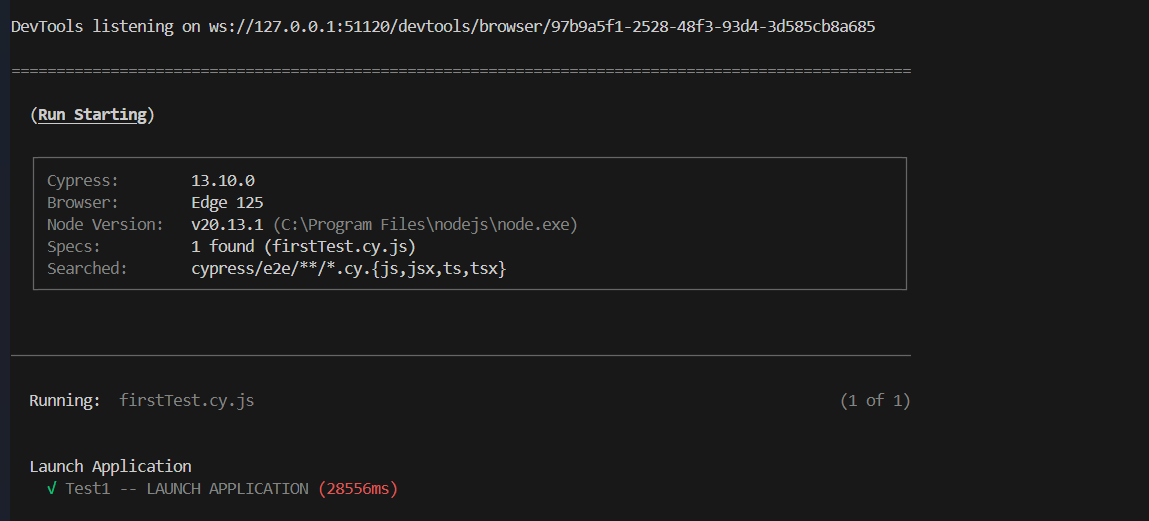
npx cypress open

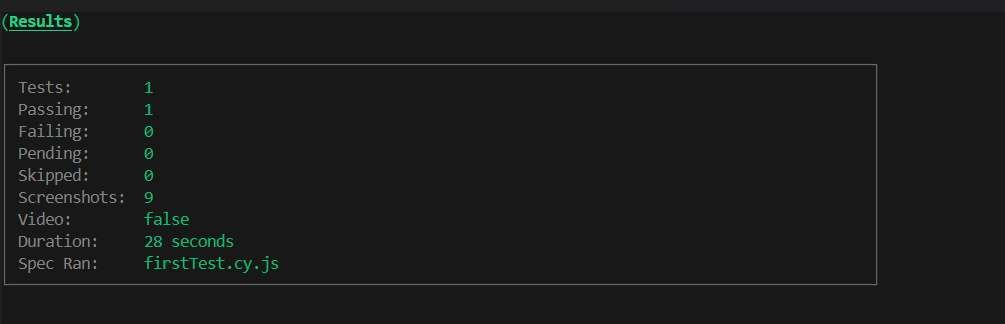
1. Select Component Testing and choose the required framework. For example: Angular

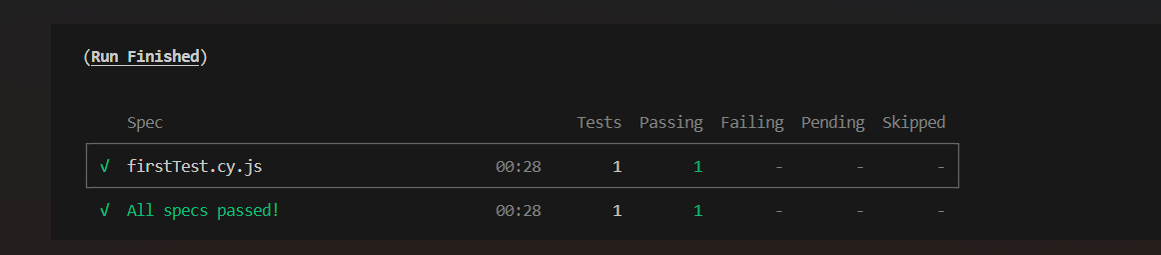
Note: If all required dependencies are installed, then “Angular” framework will be selected automatically

1. Validate all the required dependencies are installed and selected to True in next page
2. All the required cypress files are configured in the next step
3. Write a component test case under Cypress folder
4. Document Test Results:
   1. Test reports are generated using MochaAwesome plugin
   2. Screenshots for both End to End Testing and Component Testing are taken during the testing process
   3. Below are the screenshots of the console for both end to end and component testing

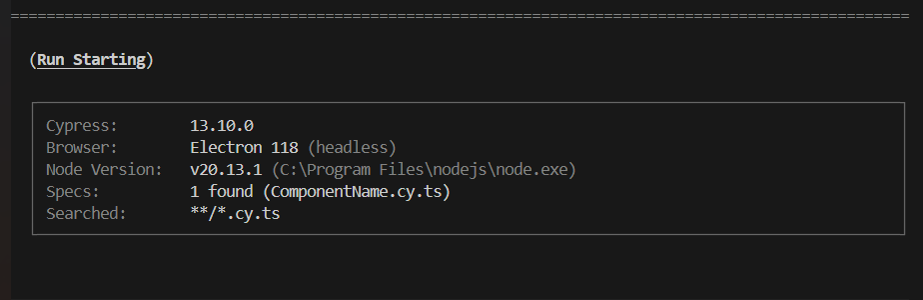
**End to End Testing**

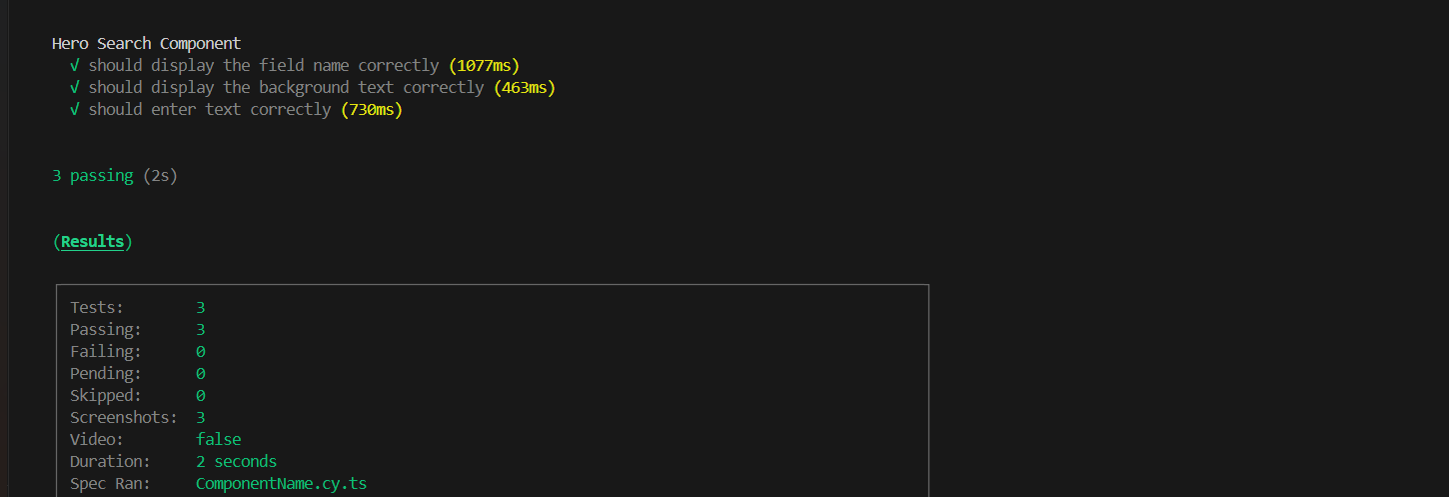






Component Testing





Note : The guide to execute test cases will be written in Readme file