

# Test-driven React.js Development: React.js Unit Testing with Enzyme and Jest

## Using Jest to Create Unit Tests

Jest is an open-source test framework created by Facebook that has a great integration with React.js. It includes a command line tool for test execution similar to what Jasmine and Mocha offer. It also allows us to create mock functions with almost zero configuration and provides a really nice set of matchers that makes assertions easier to read.


## Using Enzyme to Mount React.js Components

Enzyme provides a mechanism to mount and traverse React.js component trees. This will help us get access to its own properties and state as well as its children props in order to run our assertions.

### Test Cases

- 1) Checking the title of application to “Welcome the Matchmaking application”

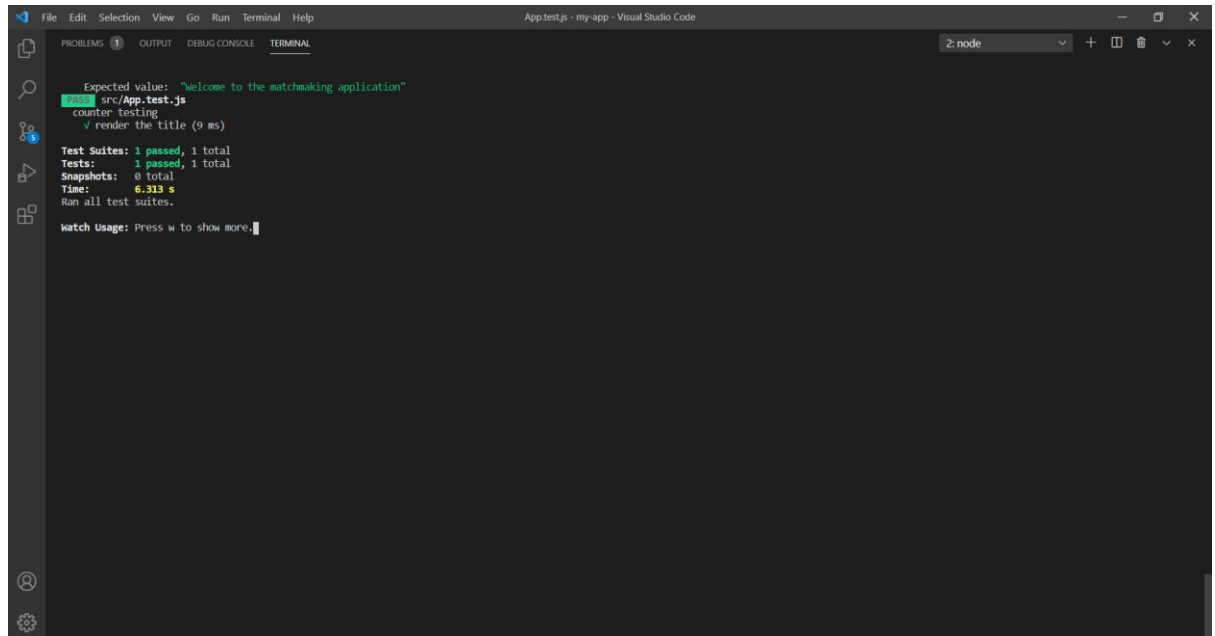
Test Case failed when we run without adding title



```
Test Suites: 1 failed, 1 total
Tests:      1 failed, 1 total
Snapshots:  0 total
Time:       8.451 s
Ran all test suites.

Watch Usage: Press w to show more.
```

Test Case passed after adding title



```
File Edit Selection View Go Run Terminal Help
App.test.js - my-app - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Expected value: "welcome to the matchmaking application"
PASS src/App.test.js
  counter testing
    ✓ render the title (9 ms)

Test Suites: 1 passed, 1 total
Tests:       1 passed, 1 total
Snapshots:  0 total
Time: 6.313 s
Ran all test suites.

Watch Usage: Press w to show more.
```

When we run the application we are getting

