

```
> questionnaire@0.1.0 main_tests /home/tg-user/CandidateCode
> jest --testTimeout=10000 --verbose --ci --testResultsProcessor="jest-junit" --
testPathPattern=main_tests/
```

FAIL main_tests/main.test.js (6.404 s)

Quiz Page

```
✓ Quiz is showing questions,options,OK and NExt button (93 ms)
× Next button is disabled & onClick Ok button next button should enables & ok should be disabled
(25 ms)
× Onclick Next button the question should change (18 ms)
× Onclick Ok button answer should be validated (46 ms)
✓ Onclick Ok button right answer should be validated (110 ms)
✓ Final Value of score (88 ms)
× Checking the timer functionality (104 ms)
```

- Quiz Page > Next button is disabled & onClick Ok button next button should enables & ok should be disabled

```
expect(element).not.toBeDisabled()
```

Received element is disabled:

```
<button data-testid="next" disabled="" />
```

```
73 |
74 |     expect(ok).toBeDisabled()
> 75 |     expect(next).not.toBeDisabled()
    |                      ^
76 |   });
77 |
78 |   test('Onclick Next button the question should change',async () => {
```

```
at call (main_tests/main.test.js:75:22)
at tryCatch (main_tests/main.test.js:2:1)
at Generator._invoke (main_tests/main.test.js:2:1)
at Generator.next (main_tests/main.test.js:2:1)
at asyncGeneratorStep (main_tests/main.test.js:2:1)
at _next (main_tests/main.test.js:2:1)
at main_tests/main.test.js:2:1
at Object.<anonymous> (main_tests/main.test.js:2:1)
```

- Quiz Page > Onclick Next button the question should change

```
expect(element).toHaveTextContent()
```

Expected element to have text content:

The use of technology to enhance learning process is called _____ in education.

Received:

A/An _____ assessment is one which measures what it is intended to measure

```
95 |     const question = screen.queryByTestId("question")
96 |     expect(question).toBeInTheDocument()
> 97 |     expect(question).toHaveTextContent("The use of technology to enhance learning process
is called _____ in education.")
    |                      ^
98 |   });
99 |
100 |   test('Onclick Ok button answer should be validated',async () => {
```

```
at call (main_tests/main.test.js:97:22)
at tryCatch (main_tests/main.test.js:2:1)
at Generator._invoke (main_tests/main.test.js:2:1)
at Generator.next (main_tests/main.test.js:2:1)
```

```

    at asyncGeneratorStep (main_tests/main.test.js:2:1)
    at _next (main_tests/main.test.js:2:1)
    at main_tests/main.test.js:2:1
    at Object.<anonymous> (main_tests/main.test.js:2:1)

```

- Quiz Page > Onclick Ok button answer should be validated

```
expect(received).toBeInTheDocument()
```

received value must be an HTMLElement or an SVGElement.
Received has value: null

```

113 |
114 |     const validate = screen.queryById("validate-answer")
> 115 |     expect(validate).toBeInTheDocument()
      |                      ^
116 |
117 |     expect(validate).toHaveTextContent("Your Answer is wrong, Correct answer is 1")
118 |

```

```

    at __EXTERNAL_MATCHER_TRAP__ (node_modules/jest-
jasmine2/node_modules/expect/build/index.js:342:30)
    at call (main_tests/main.test.js:115:22)
    at tryCatch (main_tests/main.test.js:2:1)
    at Generator._invoke (main_tests/main.test.js:2:1)
    at Generator.next (main_tests/main.test.js:2:1)
    at asyncGeneratorStep (main_tests/main.test.js:2:1)
    at _next (main_tests/main.test.js:2:1)
    at main_tests/main.test.js:2:1
    at Object.<anonymous> (main_tests/main.test.js:2:1)

```

- Quiz Page > Checking the timer functionality

```
expect(element).toHaveTextContent()
```

Expected element to have text content:

The use of technology to enhance learning process is called _____ in education.

Received:

A/An _____ assessment is one which measures what it is intended to measure

```

209 |
210 |     const question2 = screen.queryById("question")
> 211 |     expect(question2).toHaveTextContent("The use of technology to enhance learning
process is called _____ in education.")
      |                      ^
212 |
213 |     });
214 |

```

```

    at call (main_tests/main.test.js:211:23)
    at tryCatch (main_tests/main.test.js:2:1)
    at Generator._invoke (main_tests/main.test.js:2:1)
    at Generator.next (main_tests/main.test.js:2:1)
    at asyncGeneratorStep (main_tests/main.test.js:2:1)
    at _next (main_tests/main.test.js:2:1)
    at main_tests/main.test.js:2:1
    at Object.<anonymous> (main_tests/main.test.js:2:1)

```

console.error

Warning: An update to Quiz inside a test was not wrapped in act(...).

When testing, code that causes React state updates should be wrapped into act(...):

```

act(() => {
  /* fire events that update state */

```

```
});
/* assert on the output */
```

This ensures that you're testing the behavior the user would see in the browser. Learn more at <https://fb.me/react-wrap-tests-with-act>

```
in Quiz
in RenderedRoute (created by Routes)
in Routes
in Router (created by MemoryRouter)
in MemoryRouter
```

```
27 |         }
28 |       } else {
> 29 |         setTimeLeft((prev) => prev - 1);
    |         ^
30 |       }
31 |     }, 1000);
32 |   }
```

```
at printWarning (node_modules/react-dom/cjs/react-dom.development.js:88:30)
at error (node_modules/react-dom/cjs/react-dom.development.js:60:5)
at warnIfNotCurrentlyActingUpdatesInDEV (node_modules/react-dom/cjs/react-dom.development.js:23284:7)
at setTimeLeft (node_modules/react-dom/cjs/react-dom.development.js:15656:9)
at src/pages/Quiz.js:29:11
at callback (node_modules/@jest/fake-timers/build/legacyFakeTimers.js:565:32)
at FakeTimers._runTimerHandle (node_modules/@jest/fake-timers/build/legacyFakeTimers.js:601:9)
at FakeTimers.advanceTimersByTime (node_modules/@jest/fake-timers/build/legacyFakeTimers.js:292:14)
at call (main_tests/main.test.js:207:10)
at tryCatch (main_tests/main.test.js:2:1)
at Generator._invoke (main_tests/main.test.js:2:1)
at Generator.next (main_tests/main.test.js:2:1)
at asyncGeneratorStep (main_tests/main.test.js:2:1)
at _next (main_tests/main.test.js:2:1)
at main_tests/main.test.js:2:1
at Object.<anonymous> (main_tests/main.test.js:2:1)
```

Test Suites: 1 failed, 1 total

Tests: 4 failed, 3 passed, 7 total

Snapshots: 0 total

Time: 7.504 s

Ran all test suites matching /main_tests\/i.

npm ERR! code ELIFECYCLE

npm ERR! errno 1

npm ERR! questionnaire@0.1.0 main_tests: `jest --testTimeout=10000 --verbose --ci --testResultsProcessor="jest-junit" --testPathPattern=main_tests/`

npm ERR! Exit status 1

npm ERR!

npm ERR! Failed at the questionnaire@0.1.0 main_tests script.

npm ERR! This is probably not a problem with npm. There is likely additional logging output above.

npm ERR! A complete log of this run can be found in:

npm ERR! /home/tg-user/.npm/_logs/2024-06-16T08_48_57_957Z-debug.log