

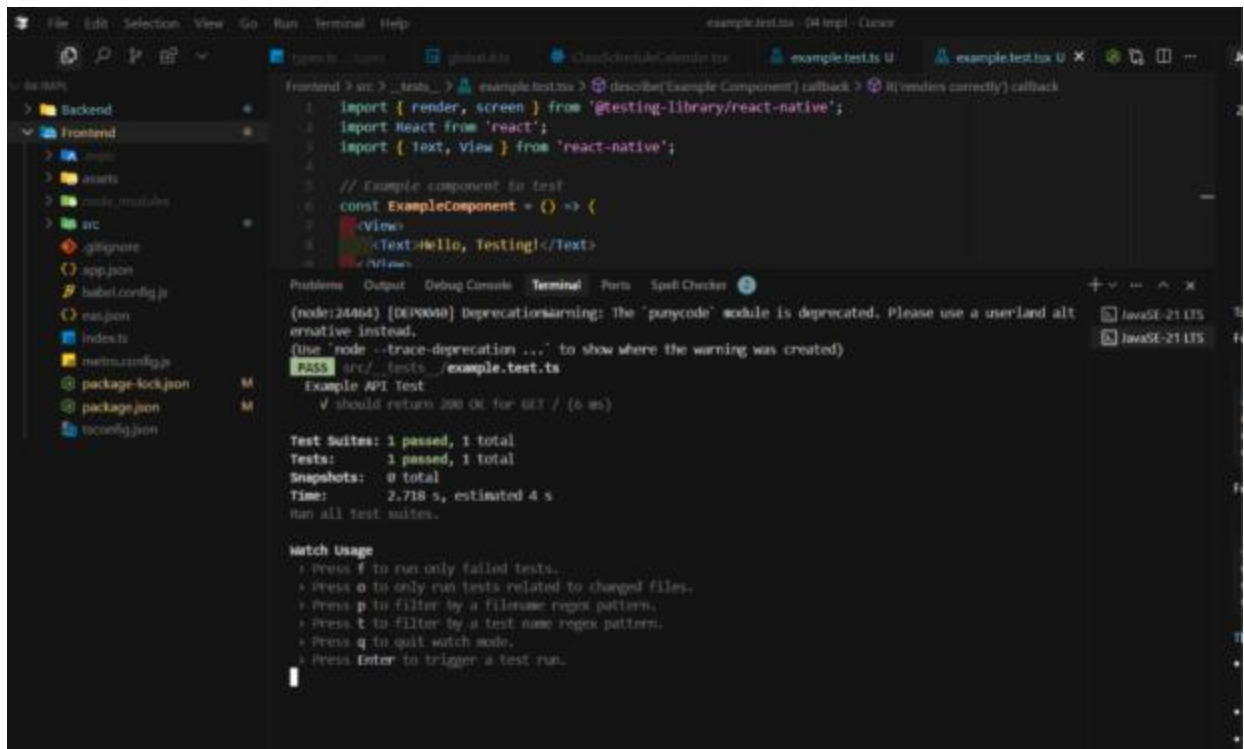
UNIT TESTING (Jest)

Frontend Testing Setup

- Test Library Used: @testing-library/react-native
- Example Component Test:
 - The test ensures that a simple ExampleComponent rendering `<Text>Hello, Testing!</Text>` works correctly.
 - Test file: Frontend/src/__tests__/example.test.tsx
 - Result: Test passed successfully (✓ renders correctly).

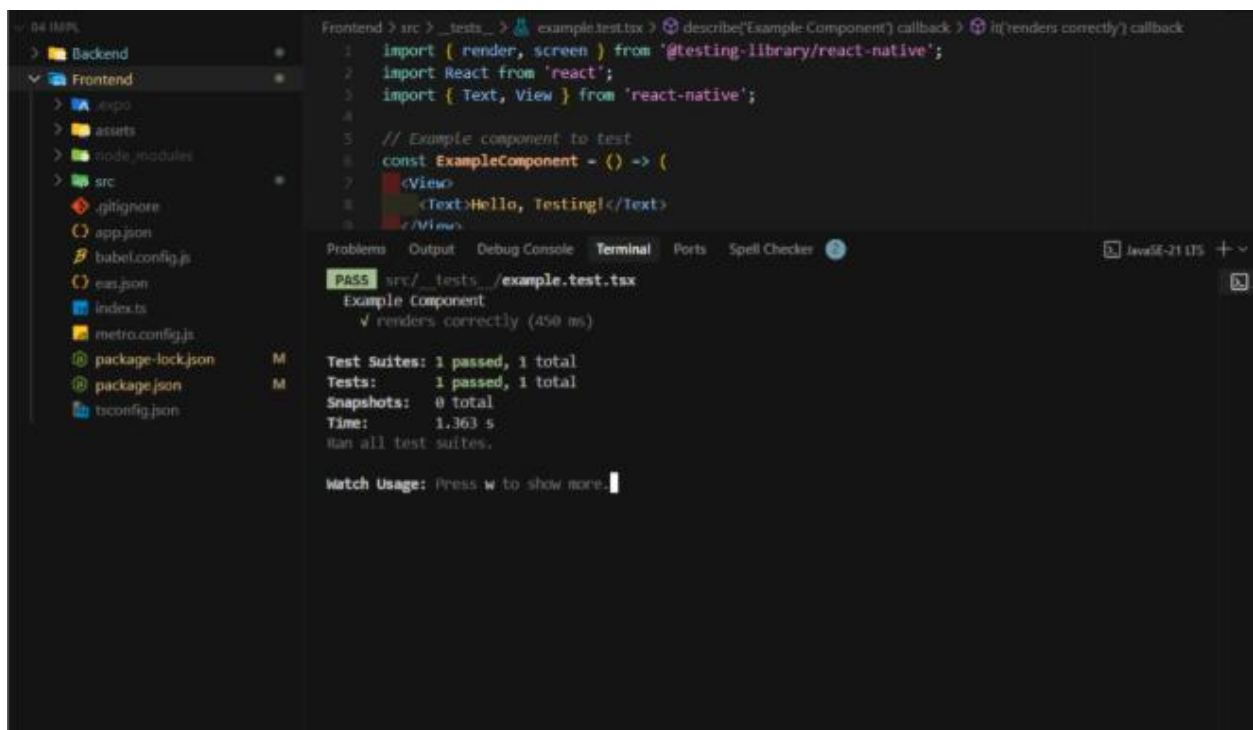
The screenshot shows an IDE with a terminal window displaying the results of a Jest unit test. The test file is `src/__tests__/example.test.tsx`, and the test passed successfully, rendering the component correctly in 4MS. Below the test results, a coverage report is displayed, showing the percentage of code covered by the tests across various files in the project.

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Line #s
All files	0	0	0	0	
src	0	100	0	0	1
config.ts	0	100	100	0	1
index.tsx	0	100	0	0	6-7
src/components	0	0	0	0	
AboutAppDrawer.tsx	0	100	0	0	18-106
ChangePasswordDrawer.tsx	0	0	0	0	23-214
ClassScheduleCalendar.tsx	0	0	0	0	45-218
ConfirmationModal.tsx	0	100	0	0	13-53
DevLoginFool.tsx	0	0	0	0	18-83
LogoutConfirmationModal.tsx	0	100	0	0	10-47
NotificationsDrawer.tsx	0	0	0	0	22-181
ProtectedRoute.tsx	0	0	0	0	5-12
SuccessModal.tsx	0	0	0	0	11-69
src/components/admin	0	0	0	0	
ActivityLogs.tsx	0	0	0	0	27-285
RecordsOverview.tsx	0	0	0	0	38-364
UserManager.tsx	0	0	0	0	21-958
src/components/instructor	0	0	0	0	
AttendanceDetails.tsx	0	0	0	0	27-294
AttendanceReports.tsx	0	0	0	0	64-774
ClassList.tsx	0	0	0	0	32-876
ClassManager.tsx	0	0	0	0	38-741
QnGenerator.tsx	0	0	0	0	17-142
QuickViewClasses.tsx	0	0	0	0	31-143
QuickViewUsers.tsx	0	0	0	0	22-125
SubjectDetailsCard.tsx	0	0	0	0	40-200
SubjectSelectionDrawer.tsx	0	0	0	0	33-189
src/components/student	0	0	0	0	
MyClasses.tsx	0	0	0	0	44-328
QnScanner.tsx	0	0	0	0	43-721
StudentList.tsx	0	0	0	0	35-116
StudentRecord.tsx	0	0	0	0	52-294
src/context	0	0	0	0	



Backend Testing Setup

- Test Library Used: supertest
- Environment: TypeScript & Node.js
- Example API Test:
 - It validates that a specific GET request returns a 200 OK response.
 - Test file: Backend/src/__tests__/example.test.ts
 - Result: Test passed successfully (✓ should return 200 OK for GET).



The screenshot shows an IDE with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with a 'Frontend' directory. The code editor displays the content of 'example.test.tsx', which imports testing utilities and defines an 'ExampleComponent'. Below the code editor, the 'Terminal' tab shows the test results, indicating that the test passed successfully.

```
Frontend > src > __tests__ > example.test.tsx > describe('Example Component') callback > it('renders correctly') callback
```

```
1 import { render, screen } from '@testing-library/react-native';
2 import React from 'react';
3 import { Text, View } from 'react-native';
4
5 // Example component to test
6 const ExampleComponent = () => {
7   <View>
8     <Text>Hello, Testing!</Text>
9   </View>
10 }
```

Problems Output Debug Console Terminal Ports Spell Checker

PASS src/ __tests__ /example.test.tsx

Example Component

✓ renders correctly (450 ms)

Test Suites: 1 passed, 1 total

Tests: 1 passed, 1 total

Snapshots: 0 total

Time: 1.363 s

Ran all test suites.

Watch Usage: Press w to show more.

