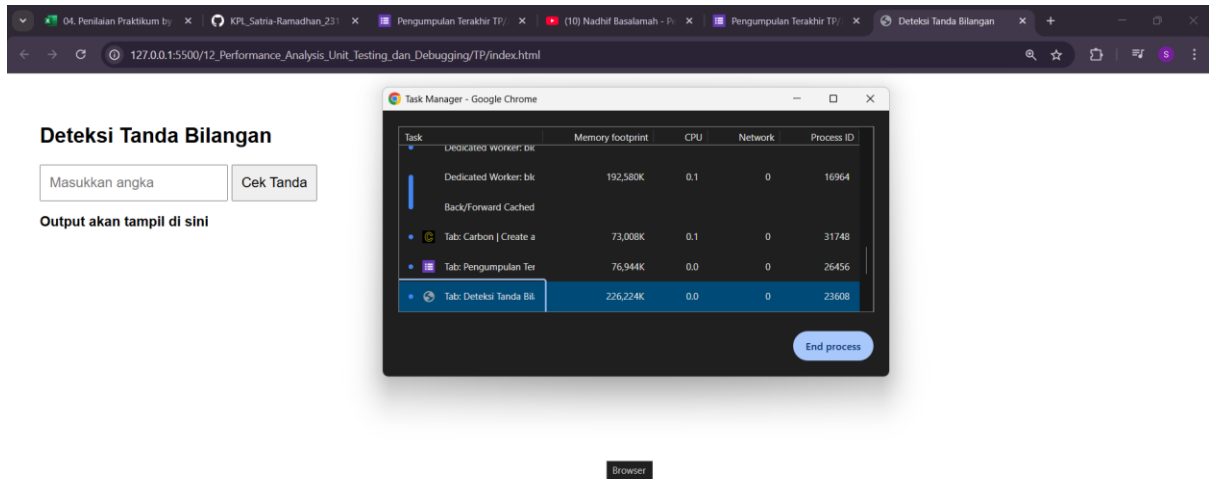


FNama : Salman Alfarisi

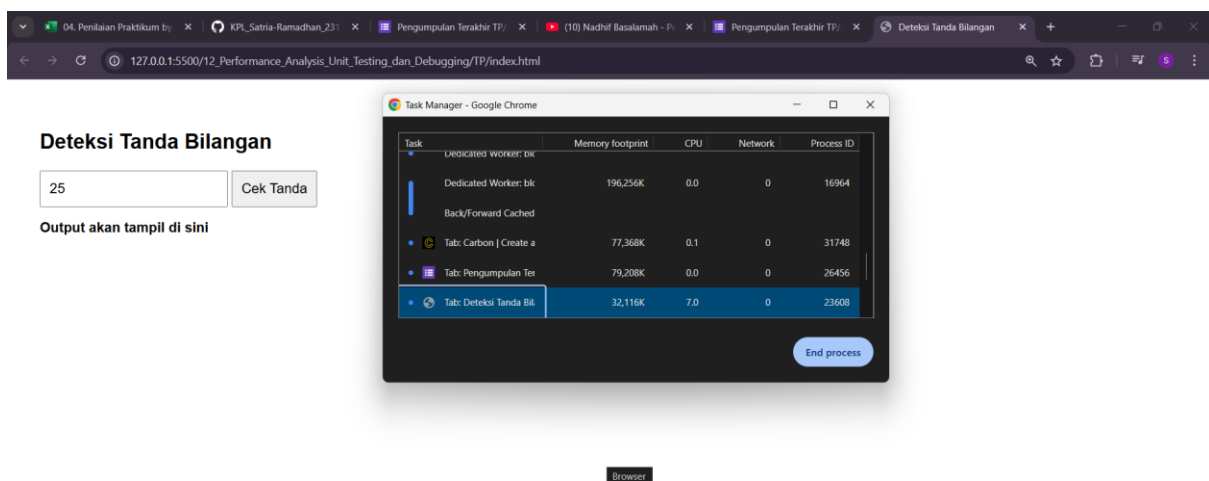
Kelas : S1SE-07-01

NIM : 2311104036

SEBELUM DIINPUT

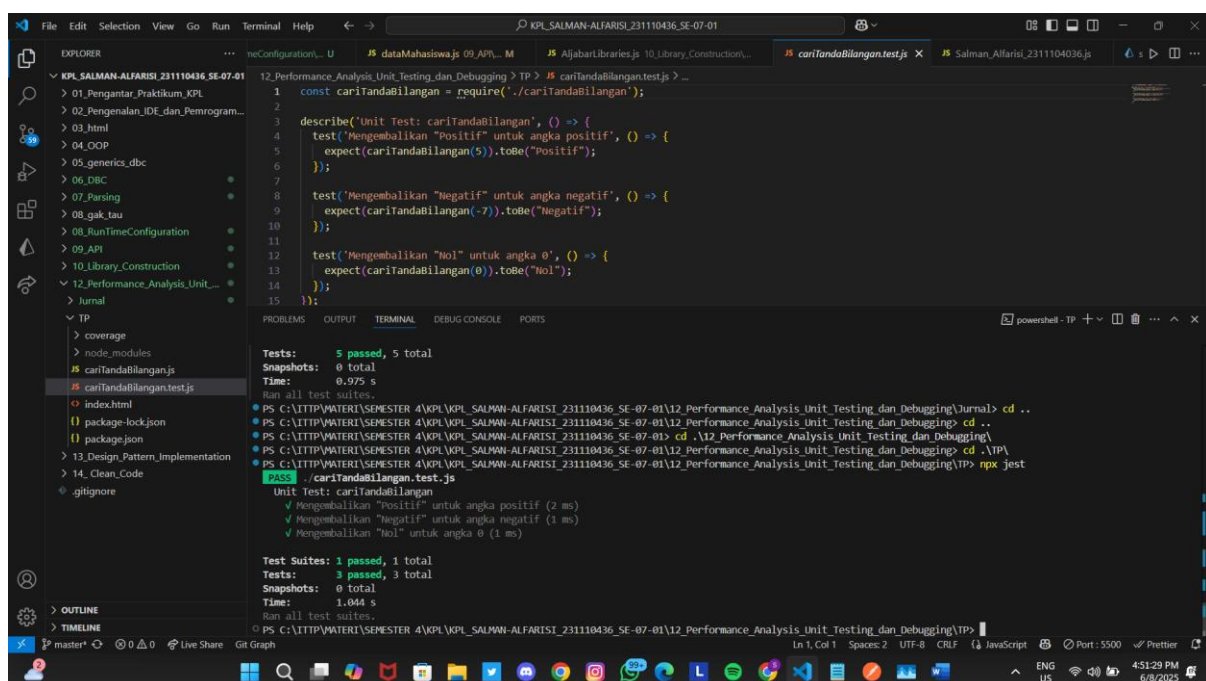


SESUDAH DIINPUT



Gambar yang ditampilkan membandingkan kondisi sebelum dan sesudah pengguna melakukan input pada aplikasi web “Deteksi Tanda Bilangan.” Sebelum pengguna memasukkan angka, terlihat bahwa penggunaan memori oleh tab “Deteksi Tanda Bilangan” adalah sebesar 226,224K, dengan penggunaan CPU sebesar 0.0%, yang menunjukkan bahwa tidak ada aktivitas komputasi yang sedang berlangsung pada saat itu. Setelah pengguna melakukan input angka, yaitu "25", dan menekan tombol “Cek Tanda”, terlihat adanya perubahan signifikan pada penggunaan CPU, yaitu naik menjadi 7.0%, sementara penggunaan memori sedikit menurun menjadi 32,116K. Hal ini menunjukkan bahwa saat input dilakukan, aplikasi memicu proses logika (kemungkinan pengecekan nilai positif, negatif, atau nol), yang mengaktifkan pemrosesan CPU secara sesaat. Penurunan penggunaan memori kemungkinan karena pembersihan atau perbaruan DOM (Document Object Model) setelah input diproses. Perubahan ini mencerminkan reaksi sistem terhadap interaksi pengguna yang memicu eksekusi fungsi logika sederhana di sisi klien.

Test npx Jest



```
1 const cariTandaBilangan = require('./cariTandaBilangan');
2
3 describe('Unit Test: cariTandaBilangan', () => {
4   test('Mengembalikan "Positif" untuk angka positif', () => {
5     expect(cariTandaBilangan(5)).toBe("Positif");
6   });
7
8   test('Mengembalikan "Negatif" untuk angka negatif', () => {
9     expect(cariTandaBilangan(-7)).toBe("Negatif");
10  });
11
12  test('Mengembalikan "Nol" untuk angka 0', () => {
13    expect(cariTandaBilangan(0)).toBe("Nol");
14  });
15 });
```

```
Tests: 5 passed, 5 total
Snapshots: 0 total
Time: 0.975 s
Run all test suites:
PS C:\ITTP\MATERI\SEMESTER 4\KPL\KPL_SALMAN-ALFARISI_231110436_SE-07-01\12_Performance_Analysis_Unit_Testing_dan_Debugging\Jurnal> cd ..
PS C:\ITTP\MATERI\SEMESTER 4\KPL\KPL_SALMAN-ALFARISI_231110436_SE-07-01\12_Performance_Analysis_Unit_Testing_dan_Debugging> cd ..
PS C:\ITTP\MATERI\SEMESTER 4\KPL\KPL_SALMAN-ALFARISI_231110436_SE-07-01> cd .\12_Performance_Analysis_Unit_Testing_dan_Debugging\
PS C:\ITTP\MATERI\SEMESTER 4\KPL\KPL_SALMAN-ALFARISI_231110436_SE-07-01\12_Performance_Analysis_Unit_Testing_dan_Debugging> cd .\TP\
PS C:\ITTP\MATERI\SEMESTER 4\KPL\KPL_SALMAN-ALFARISI_231110436_SE-07-01\12_Performance_Analysis_Unit_Testing_dan_Debugging\TP> npx jest
PASS .\cariTandaBilangan.test.js
  Unit Test: cariTandaBilangan
    ✓ Mengembalikan "Positif" untuk angka positif (2 ms)
    ✓ Mengembalikan "Negatif" untuk angka negatif (1 ms)
    ✓ Mengembalikan "Nol" untuk angka 0 (1 ms)
  Test Suites: 1 passed, 1 total
  Tests: 3 passed, 3 total
  Snapshots: 0 total
  Time: 1.004 s
  Run all test suites:
PS C:\ITTP\MATERI\SEMESTER 4\KPL\KPL_SALMAN-ALFARISI_231110436_SE-07-01\12_Performance_Analysis_Unit_Testing_dan_Debugging\TP>
```

Hasil dari pengujian unit (unit testing) pada fungsi cariTandaBilangan menggunakan framework Jest. Pada file cariTandaBilangan.test.js, dilakukan tiga pengujian untuk memastikan bahwa fungsi cariTandaBilangan memberikan output yang sesuai dengan jenis input: bilangan positif, negatif, dan nol. Hasil pengujian di terminal menunjukkan bahwa seluruh test case berhasil dijalankan dengan status “pass”. Tiga test dijalankan dengan durasi yang sangat cepat (sekitar 1–2 milidetik masing-masing), dan tidak ada snapshot yang digunakan. Keberhasilan ini menandakan bahwa logika dalam fungsi tersebut bekerja sebagaimana mestinya dan sudah meng-cover skenario input dasar dengan baik. Ini penting

dalam konteks pengembangan perangkat lunak karena menunjukkan bahwa fungsionalitas dasar telah tervalidasi secara otomatis.