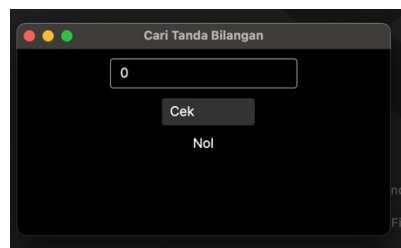


Nama: Ahmad Al – Farizi

NIM: 2311104054

- i. Link github: [https://github.com/alxfarizi/KPL\\_AHMAD-AL-FARIZI\\_2311104054\\_S1SE-07-02/tree/main/12\\_Performance\\_Analysis\\_Unit\\_Testing\\_dan\\_Debugging/TP](https://github.com/alxfarizi/KPL_AHMAD-AL-FARIZI_2311104054_S1SE-07-02/tree/main/12_Performance_Analysis_Unit_Testing_dan_Debugging/TP)
- ii. Hasil running:



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  GITLENS
- - - - -
TP git:(main) x dotnet build
Restore complete (1.2s)
tpmodul12_2311104054 succeeded (2.4s) → tpmodul12_2311104054/bin/Debug/net9.0/tpmodul12_2311104054.dll
Tests succeeded (0.7s) → Tests/bin/Debug/net9.0/Tests.dll

Build succeeded in 4.8s
TP git:(main) x dotnet test
Restore complete (1.1s)
tpmodul12_2311104054 succeeded (0.9s) → tpmodul12_2311104054/bin/Debug/net9.0/tpmodul12_2311104054.dll
[xUnit.net 00:00:00.00] xUnit.net VSTest Adapter v3.1.0+00ab0a9ec4 (64-bit .NET 9.0.5)
[xUnit.net 00:00:00.17] Discovering: tpmodul12_2311104054
[xUnit.net 00:00:00.22] Discovered: tpmodul12_2311104054
Additionally, path to test adapters can be specified using /TestAdapterPath command. Example /TestAdapterPath:<pathToCustomAdapters>.
tpmodul12_2311104054 test succeeded with 1 warning(s) (2.2s)
/usr/local/share/dotnet/sdk/9.0.300/Microsoft.TestPlatform.targets(48,5): warning : No test is available in /Users/ahmadalfarizi/TP/tpmodul12_2311104054/bin/Debug/net9.0/tpmodul12_2311104054.dll. Make sure that test discoverer & executors are registered and platform & framework version settings are appropriate and try again.
Tests succeeded (0.5s) → Tests/bin/Debug/net9.0/Tests.dll
[xUnit.net 00:00:00.00] xUnit.net VSTest Adapter v3.0.0+e341b939fe (64-bit .NET 9.0.5)
[xUnit.net 00:00:00.13] Discovering: Tests
[xUnit.net 00:00:00.23] Discovered: Tests
[xUnit.net 00:00:00.28] Starting: Tests
[xUnit.net 00:00:00.37] Finished: Tests
Tests test succeeded (2.3s)

Test summary: total: 4, failed: 0, succeeded: 4, skipped: 0, duration: 5.0s
Build succeeded with 1 warning(s) in 7.5s
TP git:(main) x
```

iii. Penjelasan source code:

Bagian antarmuka pengguna yang dibuat menggunakan XAML dalam framework Avalonia. Ada tiga komponen utama: sebuah TextBox untuk memasukkan angka, Button bertuliskan "Cek" untuk memproses input, dan TextBlock untuk menampilkan hasil. Komponen disusun secara vertikal dalam StackPanel dengan pengaturan margin dan alignment yang sederhana. File bertanggung jawab untuk menampilkan antarmuka yang memungkinkan pengguna berinteraksi dengan aplikasi.

```
1 <Window xmlns="https://github.com/avaloniaui"
2       xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
3       x:Class="tpmodul12_2311104054.MainWindow"
4       Width="400" Height="200" Title="Cari Tanda Bilangan">
5   <StackPanel Margin="10">
6       <TextBox Name="InputTextBox" Watermark="Masukkan angka" Width="200" HorizontalAlignment="Center" />
7       <Button Name="CheckButton" Content="Cek" Width="100" HorizontalAlignment="Center" Margin="10,10,0,0" Click="OnCheckButtonClick" />
8       <TextBlock Name="OutputLabel" Text="Hasil akan tampil di sini" HorizontalAlignment="Center" Margin="10" />
9   </StackPanel>
10 </Window>
11
```

Berisi kode C# yang menangani logika di balik antarmuka pengguna. Ketika tombol "Cek" diklik, metode OnCheckButtonClick akan dipanggil. Metode ini membaca input dari TextBox, mengonversinya ke bentuk integer, lalu memanggil fungsi CariTandaBilangan untuk menentukan tanda bilangan. Hasilnya ditampilkan di TextBlock. Jika input tidak valid, aplikasi akan menampilkan pesan error. File ini menjadi penghubung antara GUI dan logika bisnis.

```
1 using Avalonia.Controls;
2
3 namespace tpmodul12_2311104054
4 {
5     public partial class MainWindow : Window
6     {
7         public MainWindow()
8         {
9             InitializeComponent();
10        }
11
12        private void OnCheckButtonClick(object? sender, Avalonia.Interactivity.RoutedEventArgs e)
13        {
14            if (int.TryParse(InputTextBox.Text, out int number))
15            {
16                OutputLabel.Text = CariTandaBilangan(number);
17            }
18            else
19            {
20                OutputLabel.Text = "Input tidak valid!";
21            }
22        }
23
24        public string CariTandaBilangan(int number)
25        {
26            var helper = new BilanganHelper();
27            return helper.CariTandaBilangan(number);
28        }
29    }
30 }
31
```

Kelas ini mengandung inti logika aplikasi untuk menentukan tanda suatu bilangan. Metode CariTandaBilangan menerima parameter integer dan mengembalikan string "Positif" jika angka lebih besar dari nol, "Negatif" jika lebih kecil dari nol, dan "Nol" jika angka bernilai nol. Logika sederhana ini dipisahkan dalam kelas terpisah untuk memudahkan pengujian dan penggunaan ulang di bagian lain aplikasi.

```
1 namespace tpmodul12_2311104054
2 {
3     public class BilanganHelper
4     {
5         public string CariTandaBilangan(int number)
6         {
7             if (number > 0)
8                 return "Positif";
9             else if (number < 0)
10                 return "Negatif";
11             else
12                 return "Nol";
13         }
14     }
15 }
16
```

Berisi pengujian unit menggunakan framework Xunit untuk memverifikasi kebenaran logika di BilanganHelper. Terdapat tiga skenario pengujian: bilangan positif (5), negatif (-5), dan nol (0), masing-masing dengan hasil yang diharapkan. Pengujian ini memastikan bahwa metode CariTandaBilangan bekerja sesuai harapan dalam berbagai kasus. Dengan adanya pengujian ini, developer dapat memastikan keandalan kode meskipun terjadi perubahan di masa depan.

```
1 using Xunit;
2 using tpmodul12_2311104054;
3
4 namespace Tests
5 {
6     public class CariTandaBilanganTests
7     {
8         [Theory]
9         [InlineData(5, "Positif")]
10        [InlineData(-5, "Negatif")]
11        [InlineData(0, "Nol")]
12        public void TestCariTandaBilangan(int input, string expected)
13        {
14            var helper = new BilanganHelper();
15            var result = helper.CariTandaBilangan(input);
16            Assert.Equal(expected, result);
17        }
18    }
19 }
20
```