**Event-Driven Programming pada Aplikasi Jadwal Waktu Sholat**

Pendahuluan

Event-Driven Programming (EDP) adalah paradigma pemrograman yang mengandalkan event (peristiwa) untuk mengendalikan alur eksekusi program. Dalam paradigma ini, program tidak berjalan secara linear, tetapi menunggu suatu event terjadi, seperti klik tombol, perubahan nilai, atau peristiwa sistem.

Pada aplikasi Jadwal Waktu Sholat berbasis C# Windows Forms yang kami buat, event-driven programming menjadi fondasi utama. Setiap interaksi pengguna atau proses otomatis dikendalikan oleh event handler yang sudah didefinisikan.

Dalam artikel ini, kami akan menjelaskan secara detail setiap event, alur data, hubungan antar event, dan penjelasan kode baris per baris.

1. Event pada Tombol Back (btnBack\_Click)

Fungsi dan Alur

Tombol Back berfungsi untuk menutup form jadwal sholat dan kembali ke menu utama. Ketika tombol diklik, terjadi event Click yang akan memicu handler btnBack\_Click. Event ini penting untuk navigasi antar form.

Kode Program

private void btnBack\_Click(object sender, EventArgs e)

{

this.Close();

}

Penjelasan:

- private void btnBack\_Click(object sender, EventArgs e): Mendefinisikan metode event handler untuk event Click pada tombol btnBack.

- this.Close(): Menutup form aktif. Dengan menutup form ini, aplikasi akan kembali ke form sebelumnya.

2. Event pada Checkbox Alarm (toggleAlarm\_CheckedChanged)

Fungsi dan Alur

Checkbox digunakan untuk mengaktifkan atau menonaktifkan alarm pengingat sholat. Ketika status centang berubah, event CheckedChanged akan dipicu.

Kode Program

private void toggleAlarm\_CheckedChanged(object sender, EventArgs e)

{

if (toggleAlarm.Checked)

{

MessageBox.Show("Alarm dinyalakan");

AktifkanAlarm();

}

else

{

MessageBox.Show("Alarm dimatikan");

NonaktifkanAlarm();

}

}

Penjelasan:

- if (toggleAlarm.Checked): Mengecek apakah checkbox dalam keadaan tercentang.

- MessageBox.Show(...): Menampilkan pesan ke pengguna.

- AktifkanAlarm() / NonaktifkanAlarm(): Memanggil metode lain untuk memproses logika alarm.

3. Event Timer untuk Update Jam (timer1\_Tick)

Fungsi dan Alur

Timer berfungsi untuk memperbarui jam setiap detik secara real-time. Event Tick dipicu setiap interval timer.

Kode Program

private void timer1\_Tick(object sender, EventArgs e)

{

lblClock.Text = DateTime.Now.ToString("HH:mm:ss");

}

Penjelasan Baris Per Baris:

- lblClock.Text = ...: Mengupdate teks pada label lblClock dengan waktu saat ini, format jam:menit:detik.

4. Event Saat Form Load (Form1\_Load)

Fungsi dan Alur

Saat form pertama kali ditampilkan, event Load terjadi. Digunakan untuk menginisialisasi komponen dan menjalankan fungsi awal.

Kode Program

private void Form1\_Load(object sender, EventArgs e)

{

ShowPrayerTimes();

timer1.Start();

}

- ShowPrayerTimes(): Memanggil metode untuk menampilkan jadwal sholat.

- timer1.Start(): Menyalakan timer agar event Tick mulai bekerja.

5. Relasi Antar Event

Event Fungsi Terkait Event Lain

btnBack\_Click Navigasi ke menu Tidak

toggleAlarm\_CheckedChanged Aktif/nonaktif alarm Memanggil AktifkanAlarm/NonaktifkanAlarm

timer1\_Tick Update jam Bergantung pada timer1.Start()

Form1\_Load Inisialisasi data Memanggil ShowPrayerTimes, timer1.Start