

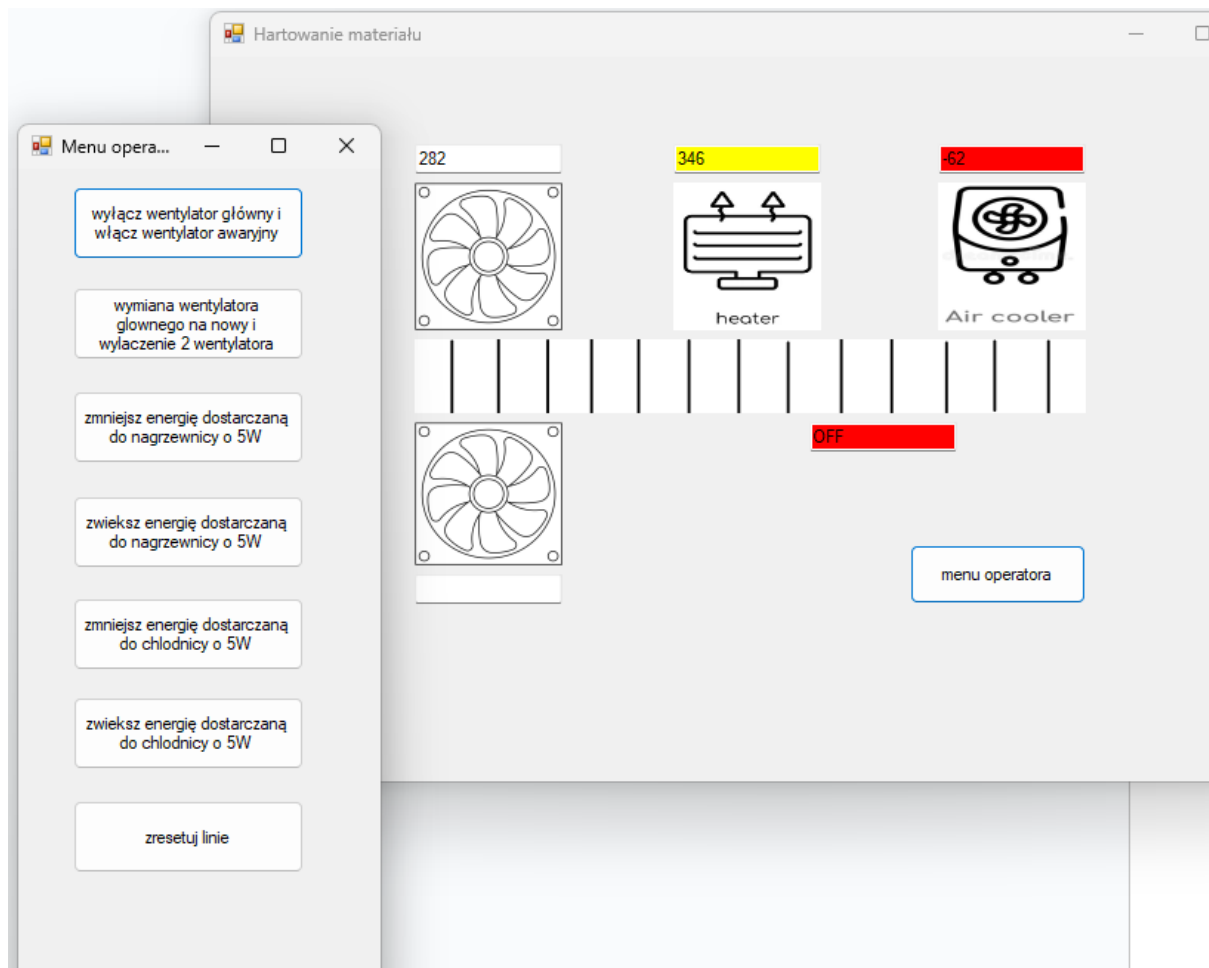
Sprawozdanie zadanie 4

“Symulacja stanowiska dyspozytorskiego”

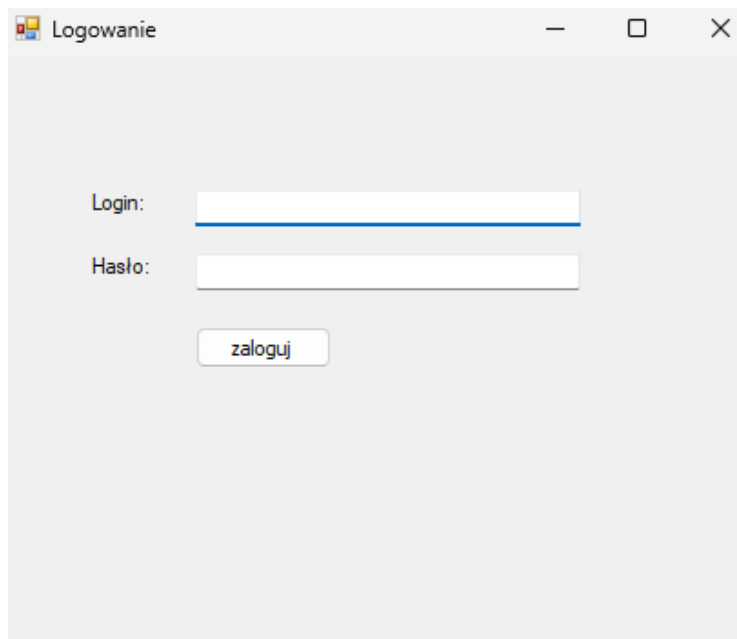
Organizacja systemów komputerowych

Adrian Nowogrodzki 184332

Oskar Nowak 184289



Program należy włączyć w trybie administratora aby udostępnić programowi informacje o parametrach podzespołów. Program symuluje pracę urządzeń przy linii produkcyjnej. Temperatura chłodnicy i nagrzewnicy zależy od temperatury CPU i GPU. Prędkość wentylatora głównego i działanie linii zależy od wygenerowanych liczb losowych. Operator ma do dyspozycji kilka działań aby wprowadzić parametry układów w odpowiednie przedziały. Nieprawidłowość parametrów sygnalizują podświetlenia na ich wyświetlaczu. Aby dostać się do opcji operatorskich należy się zalogować. Po 20 sekundach nieużywania okna operatora program prosi o ponowne logowanie.



Transferowanie danych między okienkami.

```
// import danych z innych okien przez label3
public static Form1 instance;
public Label czy_kliknieto;
1 odwołanie
public Form1()
{
    InitializeComponent();
    // import danych z innych okien
    instance = this;
    czy_kliknieto = label3;
}
```

```
Form1.instance.czy_kliknieto.Text = "tak";
```

Zamykanie okna nadzoru po upływie 20 sekund

```
if (czas_i >= 20)
{
    var formsToClose = Application.OpenForms.Cast<Form>().ToList();

    foreach (Form form in formsToClose) // iterujemy po kopii listy formularzy
    {
        if(form != this)
            form.Hide(); // zamykamy każdy formularz
    }
    if (this != null)
        this.Show();
    czas_i = 0;
}
```

Pobieranie temperatury procesora z komputera

```
using OpenHardwareMonitor.Hardware;
```

```
// Inicjalizujemy obiekt Computer
computer = new Computer();
computer.CPUEnabled = true; // Umożliwiamy odczyt temperatury procesora
computer.GPUEnabled = true; // Umożliwiamy odczyt temperatury karty graficznej
computer.FanControllerEnabled = true; // Umożliwiamy odczyt prędkości wentylatorów
computer.MainboardEnabled = true; // Umożliwiamy odczyt temperatury płyty głównej

computer.Open();
```

```
// nagrzewnica
// Przeszukujemy wszystkie obiekty w systemie, w tym CPU
foreach (var hardwareItem in computer.Hardware)
{
    hardwareItem.Update();

    // Szukamy CPU
    if (hardwareItem.HardwareType == HardwareType.CPU)
    {
        // Dla CPU pobieramy temperature
        foreach (var sensor in hardwareItem.Sensors)
        {
            if (sensor.SensorType == SensorType.Temperature)
            {
                nagrzewnica_i = 300 + (int)sensor.Value + nagrzewnica_inger_oper_i;
                nagrzewnica_tb.Text = nagrzewnica_i.ToString();
            }
        }
    }
}
```

Działanie przycisku operatora "wyłącz wentylator główny i włącz awaryjny"

```
// ingerencja operatora:
if (ingerencja_operator_s.Text == "wylacz wentylator glowny i włącz awaryjny")
{
    went1_on_off = false;
    wentylator_tb.BackColor = Color.White;
    wentylator_tb.Text = "0";

    went2_on_off = true;
}
```