Generator de figuri

Descrierea programului:

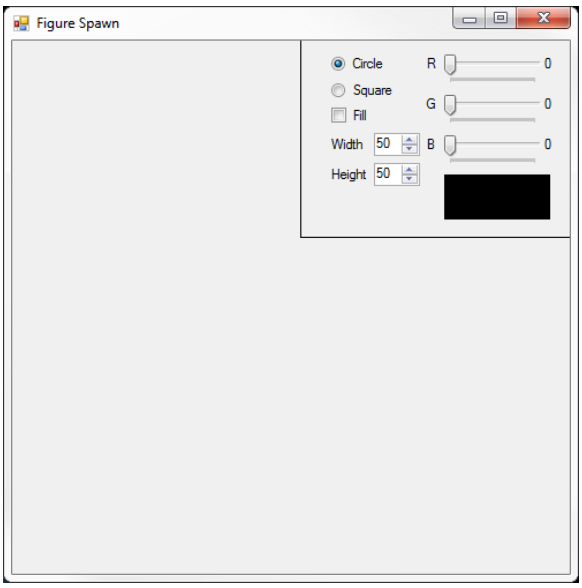
Programul a fost realizat in Windows Forms Application utilizand Microsoft Visual Studio 2015. Acesta contine urmatoarele elemente:

- 2 “Radio buttons” folosite pentru a selecta figura dorita (elipsa sau dreptunghi)
- 1 “Check box” utilizat pentru a alege daca figura sa fie umpluta sau goala
- 3 “Track bars” folosite la ajustarea culorii figurii
- 1 “Picture box” ce afiseaza culoarea curenta
- 2 “Numeric up down” care au rolul de a ajusta dimensiunea figurii (latimea si lungimea)
- 11 etichete folosite pentru a indica instrumentele de mai sus sau valorile utilizate

Programul este folosit pentru a genera o figura (elipsa sau dreptunghi). Utilizatorul are la dispozitie mai multe unelte pentru a modifica culoarea, dimensiunea si optiunea de a alege daca figura sa fie colorata in intregime sau doar conturul acesteia.

Initial la pornirea programului cele 3 valori “RGB” folosite la generarea culorii sunt 0, latimea si lungimea figurii au valoarea 50, optiunea de a colora figura in intregime este dezactivata iar figura aleasa initial este elipsa.

Fereastra initiala a programului:



Exemple:

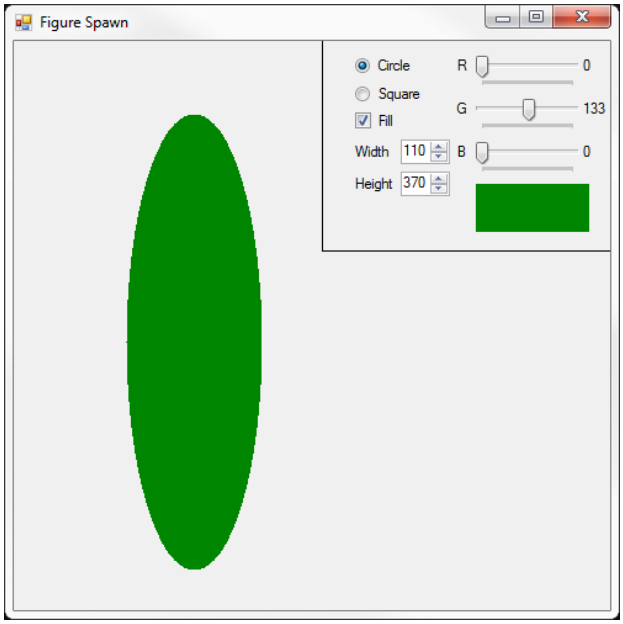


Figure Spawn
Circle
Square
Fill
Width 110
Height 370
R 0
G 133
B 0

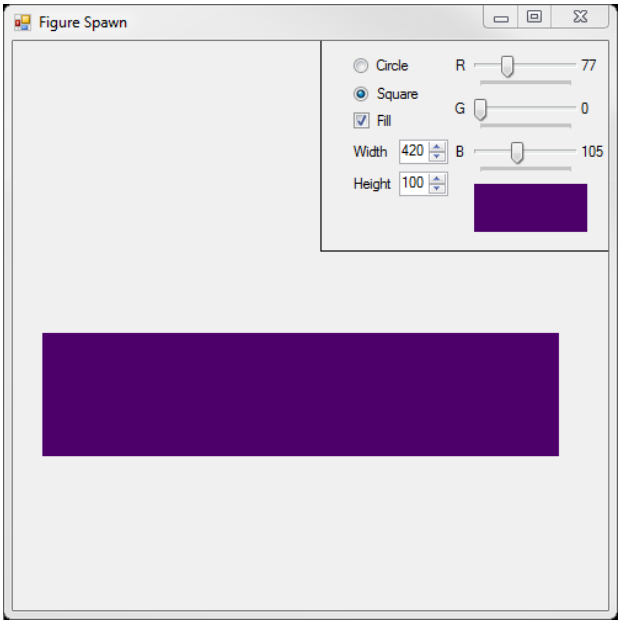


Figure Spawn
Circle
Square
Fill
Width 420
Height 100
R 77
G 0
B 105

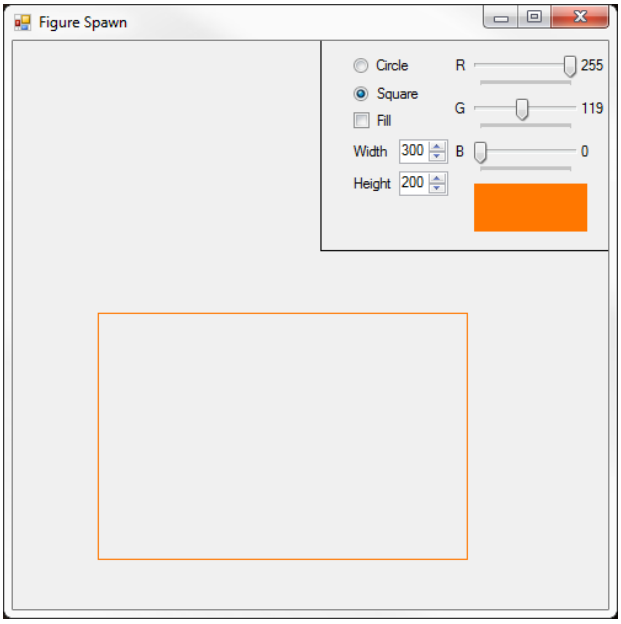


Figure Spawn
Circle
Square
Fill
Width 300
Height 200
R 255
G 119
B 0

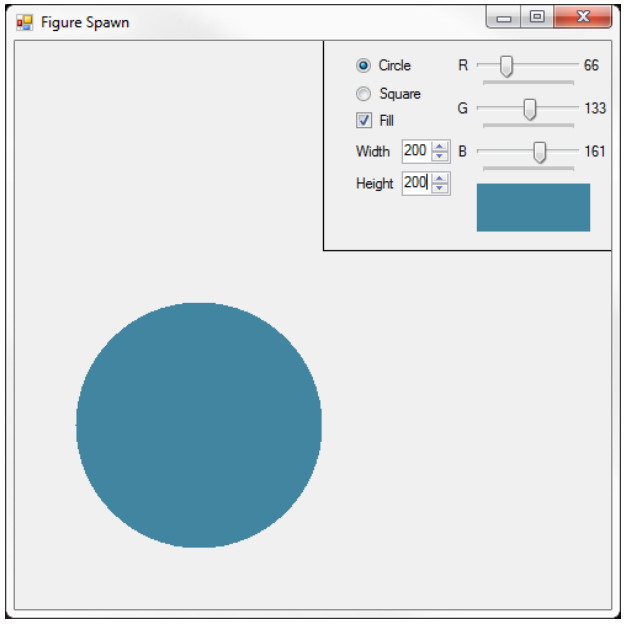


Figure Spawn
Circle
Square
Fill
Width 200
Height 200
R 66
G 133
B 161

Codul Sursa:

```
        bool init = false;
        int click_x = 0, click_y = 0;
        int col_R, col_G, col_B;

        private void Form1_MouseMove(object sender, MouseEventArgs e)
        {
            click_x = e.X;
            click_y = e.Y;
            //this.Refresh();
        }

        private void Form1_Paint(object sender, PaintEventArgs e)
        {
            this.numericUpDown1.Location = new System.Drawing.Point(this.Width - 186, 80);
            this.numericUpDown2.Location = new System.Drawing.Point(this.Width - 186, 106);
            this.pictureBox1.Location = new System.Drawing.Point(this.Width - 125, 116);
            this.radioButton1.Location = new System.Drawing.Point(this.Width - 223, 11);
            this.radioButton2.Location = new System.Drawing.Point(this.Width - 223, 34);
            this.trackBar1.Location = new System.Drawing.Point(this.Width - 133, 10);
            this.trackBar2.Location = new System.Drawing.Point(this.Width - 133, 45);
            this.trackBar3.Location = new System.Drawing.Point(this.Width - 133, 80);
            this.checkBox1.Location = new System.Drawing.Point(this.Width - 223, 57);
            this.label1.Location = new System.Drawing.Point(this.Width - 143, 13);
            this.label2.Location = new System.Drawing.Point(this.Width - 143, 48);
            this.label3.Location = new System.Drawing.Point(this.Width - 143, 83);
            this.label4.Location = new System.Drawing.Point(this.Width - 41, 13);
            this.label5.Location = new System.Drawing.Point(this.Width - 41, 48);
            this.label6.Location = new System.Drawing.Point(this.Width - 41, 83);
            this.label7.Location = new System.Drawing.Point(this.Width - 226, 83);
            this.label8.Location = new System.Drawing.Point(this.Width - 226, 109);
            System.Drawing.Graphics Drawing;
            Drawing = this.CreateGraphics();
            System.Drawing.SolidBrush Brush_Custom;
            System.Drawing.Pen Pen_Custom;
            System.Drawing.Pen Pen_Black;
            Brush_Custom =
            new System.Drawing.SolidBrush(System.Drawing.Color.FromArgb(col_R, col_G, col_B));
            Pen_Custom = new System.Drawing.Pen(System.Drawing.Color.FromArgb(col_R, col_G, col_B));
            Pen_Black = new System.Drawing.Pen(System.Drawing.Color.Black);
            int figure_width = 0, figure_height = 0;
            figure_width = Convert.ToInt16(numericUpDown1.Value);
            figure_height = Convert.ToInt16(numericUpDown2.Value);
            Rectangle tool_zone = new Rectangle(this.Width - 250, -5, 250, 175);
            Rectangle drawing_restricted_zone =
            new Rectangle(this.Width - 250 - (figure_width / 2), -5, 250 + (figure_width / 2), 175 +
            (figure_height / 2));
            Drawing.DrawRectangle(Pen_Black, tool_zone);
            //Drawing.DrawRectangle(Pen_Black, drawing_restricted_zone); linie folosita pentru a testa
zona de restrictionare a generarii figurii
            if (drawing_restricted_zone.Contains(this.PointToClient(Cursor.Position)) && (init))
```

```
    {
        init = false;
    }

    if (!drawing_restricted_zone.Contains(this.PointToClient(Cursor.Position)) && (init))
    {
        if ((this.radioButton1.Checked == true) && (this.checkBox1.Checked == false))
        {
            Drawing.DrawEllipse(Pen_Custom, click_x - figure_width/2, click_y –
            figure_height/2, figure_width, figure_height);
        }
        else if ((this.radioButton1.Checked == true) && (this.checkBox1.Checked == true))
            {
                Drawing.FillEllipse(Brush_Custom, click_x - figure_width / 2, click_y –
                figure_height / 2, figure_width, figure_height); }
            }

    if (!drawing_restricted_zone.Contains(this.PointToClient(Cursor.Position)) && (init))
    {
        if ((this.radioButton2.Checked == true) && (this.checkBox1.Checked == false))
        {
            Drawing.DrawRectangle(Pen_Custom, click_x - figure_width / 2, click_y –
            figure_height / 2, figure_width, figure_height);
        }
        else if ((this.radioButton2.Checked == true) && (this.checkBox1.Checked == true))
            {
                Drawing.FillRectangle(Brush_Custom, click_x - figure_width / 2, click_y -
                figure_height / 2, figure_width, figure_height); }
            }

}

private void checkBox2_MouseClick(object sender, MouseEventArgs e)
{
    init = true;
    this.Refresh();
}

private void trackBar1_Scroll(object sender, EventArgs e)
{
    col_R = this.trackBar1.Value;
    this.pictureBox1.BackColor = System.Drawing.Color.FromArgb(col_R, col_G, col_B);
    this.label4.Text = col_R.ToString();
    this.label4.Refresh();
}

private void trackBar2_Scroll(object sender, EventArgs e)
{
    col_G = this.trackBar2.Value;
    this.pictureBox1.BackColor = System.Drawing.Color.FromArgb(col_R, col_G, col_B);
    this.label5.Text = col_G.ToString();
    this.label5.Refresh();
}


private void trackBar3_Scroll(object sender, EventArgs e)
{
    col_B = this.trackBar3.Value;
    this.pictureBox1.BackColor = System.Drawing.Color.FromArgb(col_R, col_G, col_B);
    this.label6.Text = col_B.ToString();
    this.label6.Refresh();
```

```
}

private void numericUpDown1_ValueChanged(object sender, EventArgs e)
{
    if (this.numericUpDown1.Value > 991)
    {
        this.numericUpDown1.Increment = this.numericUpDown1.Value-990;
    }

    if (this.numericUpDown1.Value <= 990)
    {
        this.numericUpDown1.Increment = 10;
    }
}

private void numericUpDown2_ValueChanged(object sender, EventArgs e)
{
    if (this.numericUpDown2.Value > 991)
    {
        this.numericUpDown2.Increment = this.numericUpDown2.Value - 990;
    }

    if (this.numericUpDown2.Value <= 990)
    {
        this.numericUpDown2.Increment = 10;
    }
}

private void Form1_Resize(object sender, EventArgs e)
   {
    init = false;
    this.Refresh();
   }
```