

Laporan Pembuatan Aplikasi Manajemen Marketplace PETONA



Oleh

Nazwa Dafa Ramadhan Arifin Putra	20051397023
Mohammad Dzakiyyul Ashfiya' El Arif	20051397031
Gagah Rizky Mulyawan	20051397045

Program Studi Manajemen Informatika

Fakultas Vokasi

Universitas Negeri Surabaya

2022

Deskripsi Aplikasi

OPO HALLLLLLL LLLL LLLL LLLL LLLL LLLL LLLL LLLL

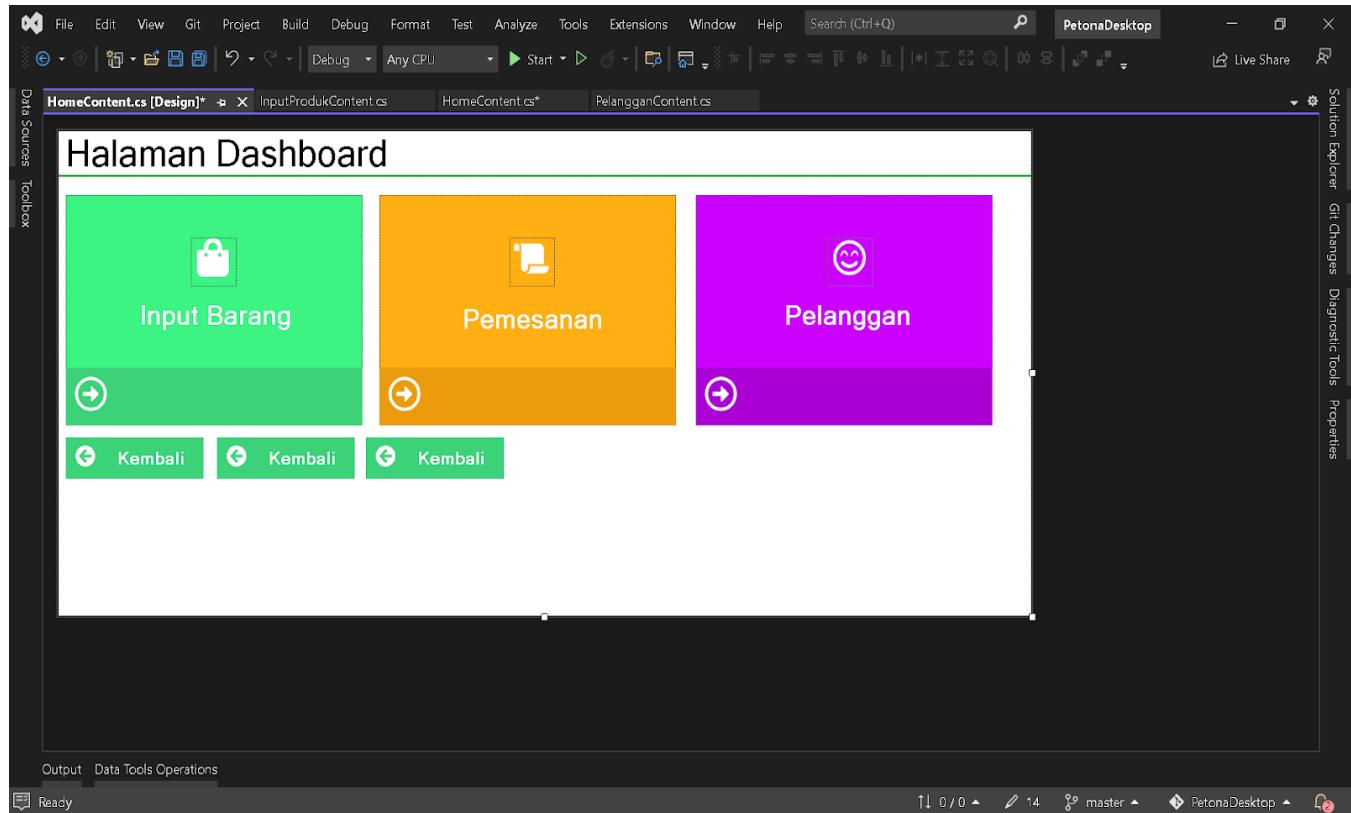
Teknis penggunaan aplikasi

Jelasno ges

Kontribusi Setiap Anggota

1. Nazwa Dafa Ramadhan Arifin Putra

Membuat halaman home yang berisikan 3 pilihan yaitu Input Barang, Pemesanan, dan Pelanggan



Inisialisasi Posisi dan ukuran konten Input Produk, Pemesanan, dan Pelanggan

```
10
11     namespace PetonaDesktop
12     {
13         public partial class HomeContent : UserControl
14         {
15             public HomeContent()
16             {
17                 InitializeComponent();
18
19                 // inisialisasi posisi dan ukuran dari konten input produk, pemesanan, pelanggan
20                 InputProdukContent.SendToBack();
21                 InputProdukContent.Location = new Point(0,0);
22                 InputProdukContent.Size = new Size(1600, 1145);
23
24                 PelangganContent.SendToBack();
25                 PelangganContent.Location = new Point(0,0);
26                 PelangganContent.Size = new Size(1600,1145);
27
28                 PemesananContent.SendToBack();
29                 PemesananContent.Location = new Point(0,0);
30                 PemesananContent.Size = new Size(1600,1145);
31
32             }
33         }
34     }
```

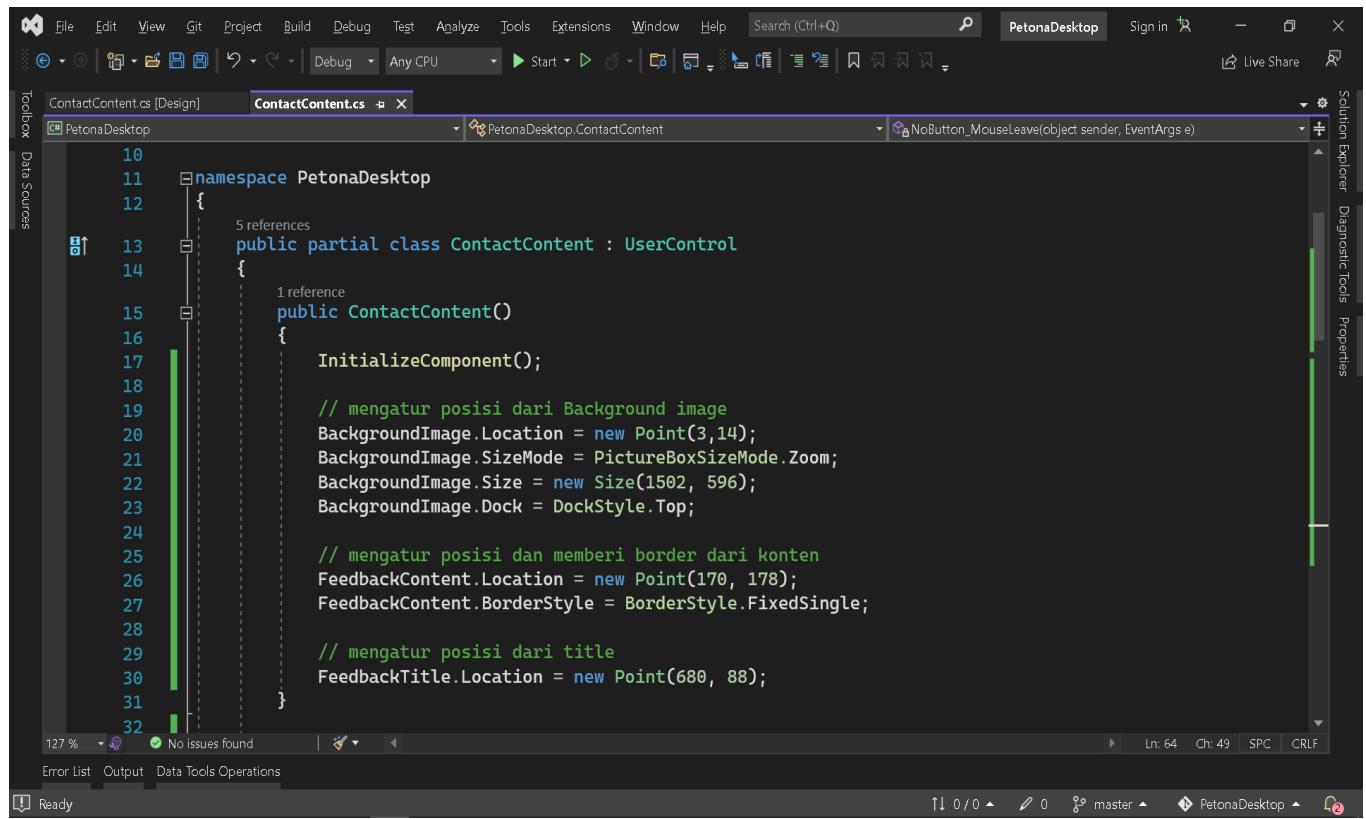
Ketika panel Input barang, Pemesanan, atau Pelanggan ditekan maka akan membuka masing-masing tampilan dengan megakses user control baru

```
private void InputBarangButton_Click(object sender, EventArgs e)
{
    // tampilkan InputProdukContent
    InputProdukContent.BringToFront();
}

private void PemesananButton_Click(object sender, EventArgs e)
{
    // tampilkan PemesananContent
    PemesananContent.BringToFront();
}

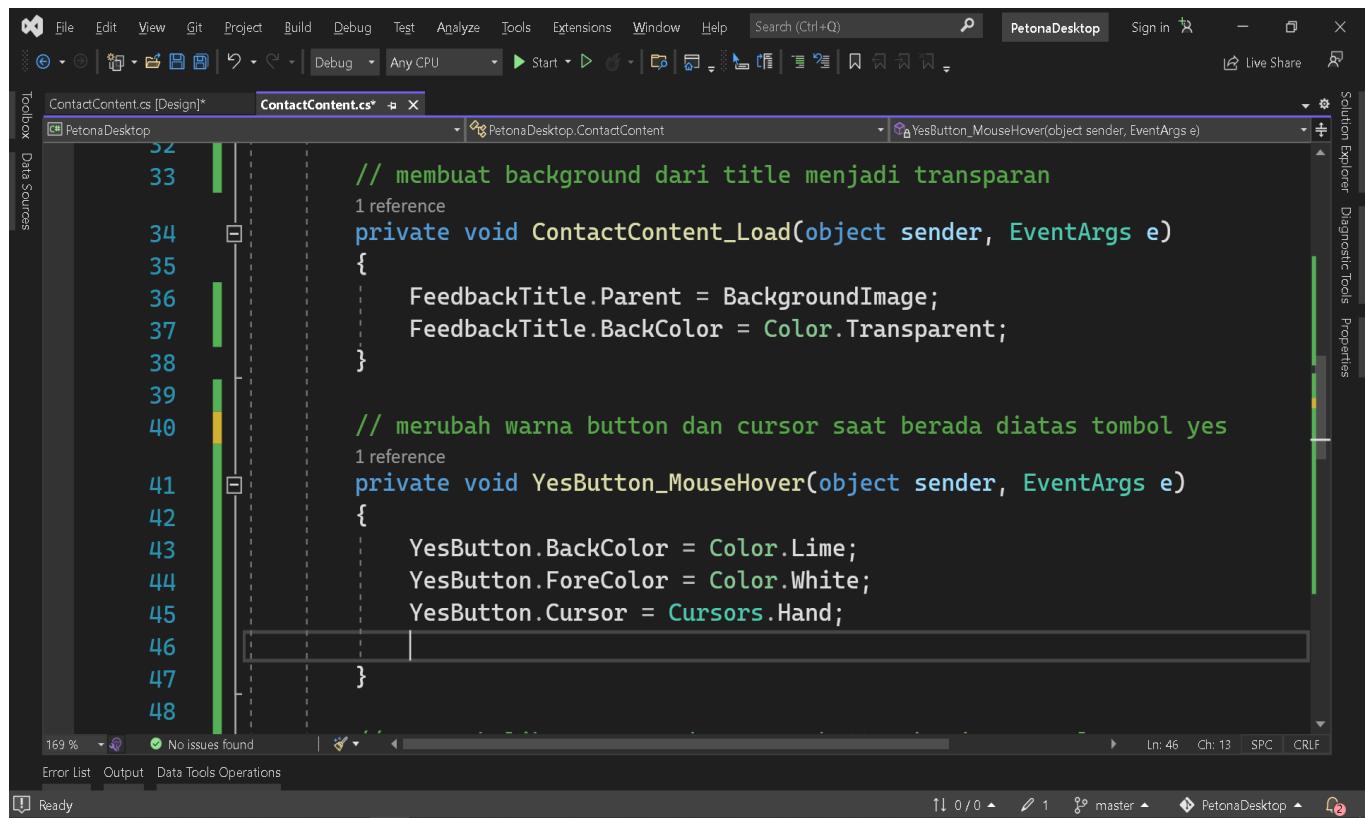
private void CustomerButton_Click(object sender, EventArgs e)
{
    // tampilkan PelangganContent
    PelangganContent.BringToFront();
}
```

2. Gagah Rizky Mulyawan



Screenshot of Visual Studio showing the ContactContent.cs code. The code defines a partial class ContactContent that initializes its components and sets their properties.

```
10
11  namespace PetonaDesktop
12  {
13      public partial class ContactContent : UserControl
14      {
15          public ContactContent()
16          {
17              InitializeComponent();
18
19              // mengatur posisi dari Background image
20              BackgroundImage.Location = new Point(3, 14);
21              BackgroundImage.SizeMode = PictureBoxSizeMode.Zoom;
22              BackgroundImage.Size = new Size(1502, 596);
23              BackgroundImage.Dock = DockStyle.Top;
24
25              // mengatur posisi dan memberi border dari konten
26              FeedbackContent.Location = new Point(170, 178);
27              FeedbackContent.BorderStyle = BorderStyle.FixedSingle;
28
29              // mengatur posisi dari title
30              FeedbackTitle.Location = new Point(680, 88);
31
32      }
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
```



Screenshot of Visual Studio showing the ContactContent.cs code with additional logic. It includes methods for handling the load event of the control and the mouse hover event of a button.

```
52
53  // membuat background dari title menjadi transparan
54
55  private void ContactContent_Load(object sender, EventArgs e)
56  {
57      FeedbackTitle.Parent = BackgroundImage;
58      FeedbackTitle.BackColor = Color.Transparent;
59
60
61  // merubah warna button dan cursor saat berada diatas tombol yes
62  private void YesButton_MouseHover(object sender, EventArgs e)
63  {
64      YesButton.BackColor = Color.Lime;
65      YesButton.ForeColor = Color.White;
66      YesButton.Cursor = Cursors.Hand;
67
68
69
```

The screenshot shows the Visual Studio IDE interface with the following details:

- File Menu:** File, Edit, View, Git, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help.
- Search Bar:** Search (Ctrl+Q).
- Solution Explorer:** PetonaDesktop.
- Toolbox:** Toolbox, Data Sources.
- Code Editor:** ContactContent.cs [Design]*, ContactContent.cs*. The code is as follows:

```
// mengembalikan cursor dan warna button keadaan semula
private void YesButton_MouseLeave(object sender, EventArgs e)
{
    YesButton.BackColor = Color.White;
    YesButton.ForeColor = Color.Black;
}

// merubah warna dan cursor saat berada diatas tombol no
private void NoButton_MouseHover(object sender, EventArgs e)
{
    NoButton.BackColor = Color.Lime;
    NoButton.ForeColor = Color.White;
    NoButton.Cursor = Cursors.Hand;
}
```

Status Bar: 186%, No issues found, Error List, Output, Data Tools Operations, Ready.

The screenshot shows the Visual Studio IDE interface with the following details:

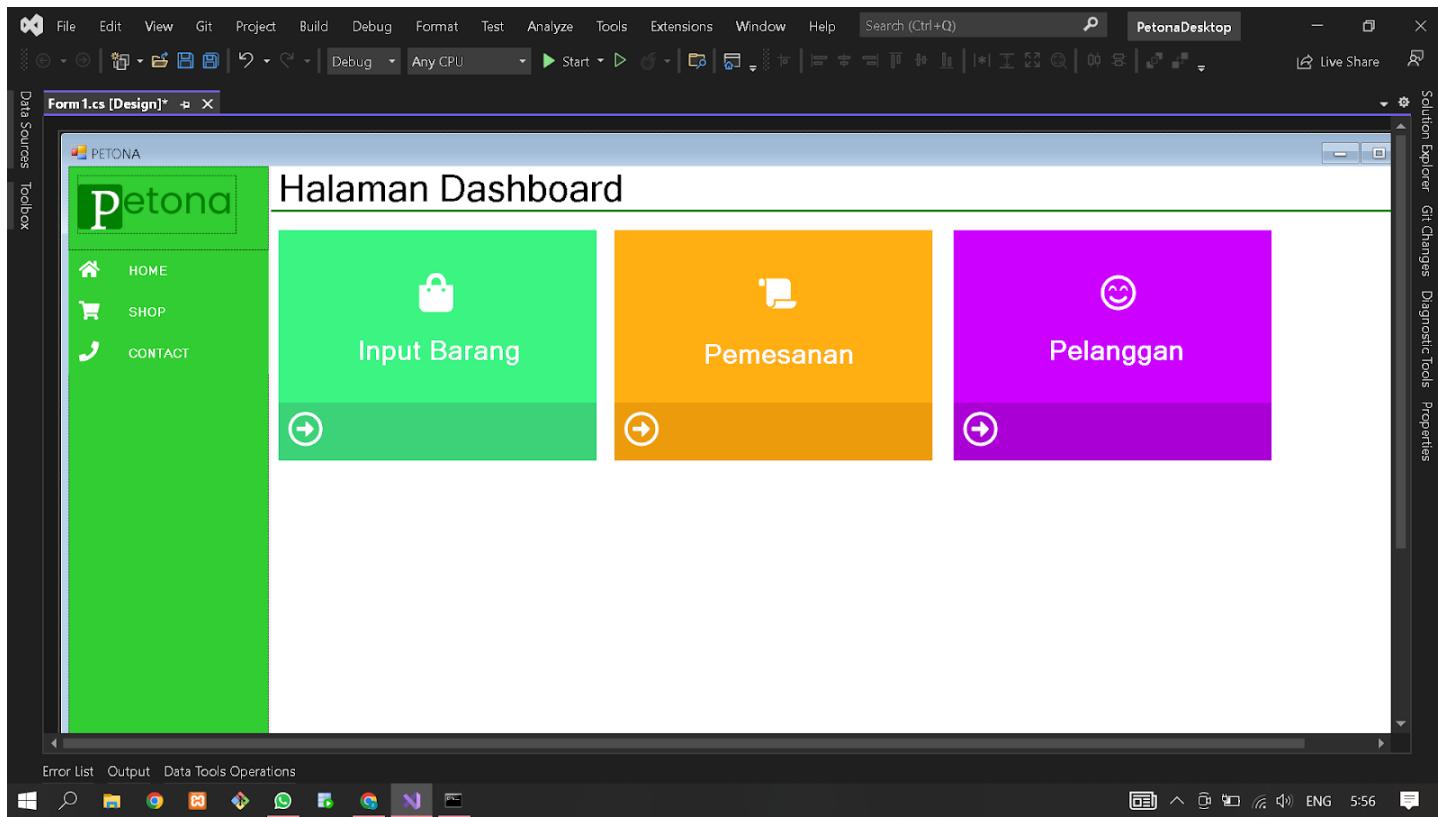
- File Menu:** File, Edit, View, Git, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help.
- Search Bar:** Search (Ctrl+Q).
- Solution Explorer:** PetonaDesktop.
- Toolbox:** Toolbox, Data Sources.
- Code Editor:** ContactContent.cs [Design]*, ContactContent.cs*. The code is as follows:

```
// mengembalikan cursor dan warna button keadaan semula
private void NoButton_MouseLeave(object sender, EventArgs e)
{
    NoButton.BackColor = Color.White;
    NoButton.ForeColor = Color.Black;
}
```

Status Bar: 186%, No issues found, Error List, Output, Data Tools Operations, Ready.

3. Mohammad Dzakiyyul Ashfiya' El Arif

Membuat Navbar Panel yang berisikan 1 picturebox dan 3 button untuk Home, Shop, dan Contact



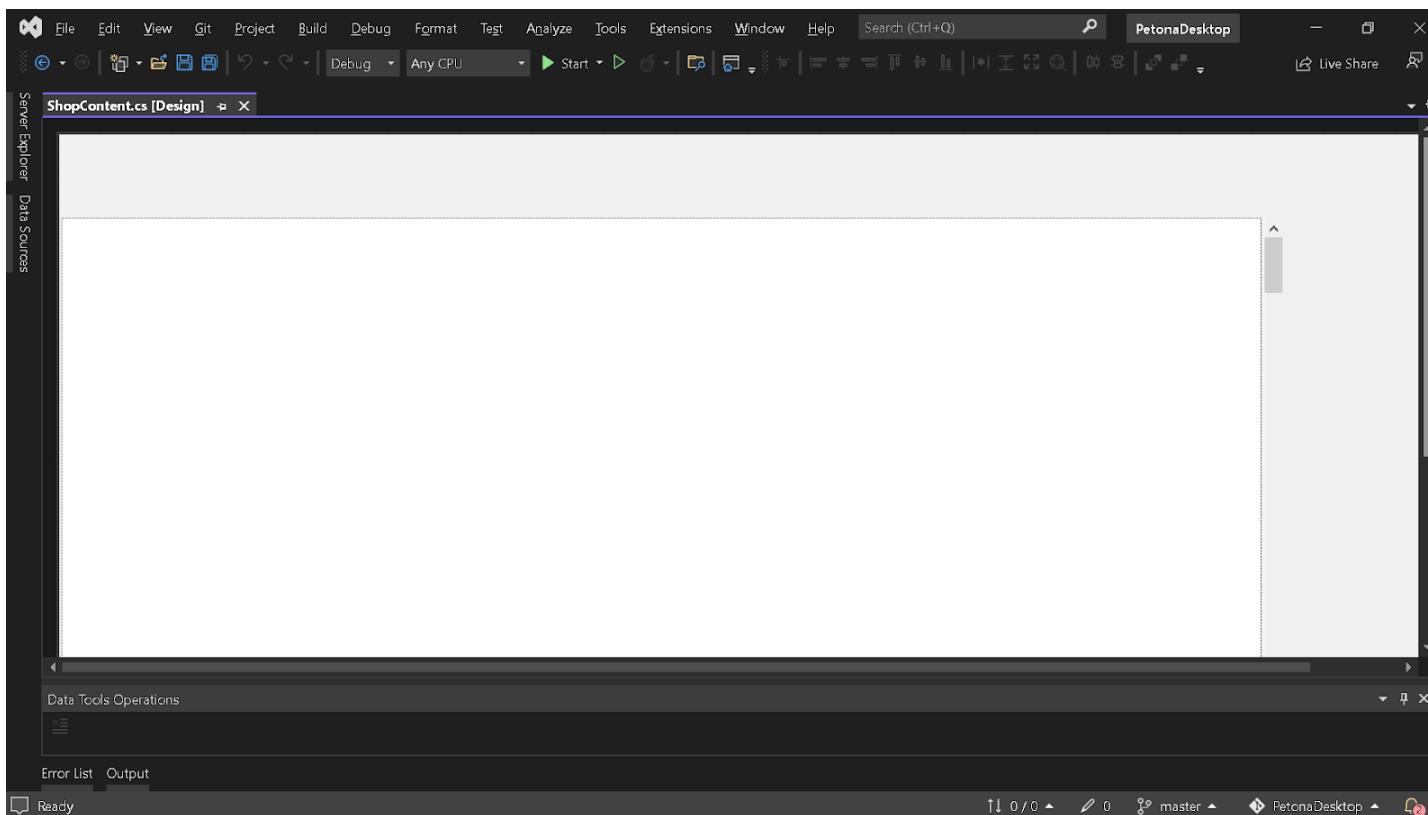
Inisialisasi posisi awal setiap konten agar posisi dan ukuran konten memenuhi tepat disamping navbar panel.

```
4 using System.Data;
5 using System.Drawing;
6 using System.Linq;
7 using System.Text;
8 using System.Threading.Tasks;
9 using System.Windows.Forms;
10
11 namespace PetonaDesktop
12 {
13     public partial class Form1 : Form
14     {
15         public Form1()
16         {
17             InitializeComponent();
18
19             // inisialisasi posisi awal setiap konten
20             ShopContent.Location = new Point(270, 0);
21             ShopContent.Size = new Size(1650, 1145);
22
23             ContactContent.Location = new Point(270, 0);
24             ContactContent.Size = new Size(1650, 1145);
25
26             HomeContent.BringToFront();
27             HomeContent.Location = new Point(270, 0);
28             HomeContent.Size = new Size(1650, 1145);
29     }
30
31     // menempatkan konten ke posisi paling depan
32     private void ShopBtn_Click(object sender, EventArgs e)
33     {
34         ShopContent.BringToFront();
35     }
36
37     // menempatkan konten ke posisi paling depan
38     private void ContactBtn_Click(object sender, EventArgs e)
39     {
40         ContactContent.BringToFront();
41     }
42
43     // menempatkan konten ke posisi paling depan
44     private void HomeBtn_Click(object sender, EventArgs e)
45     {
46         HomeContent.BringToFront();
47     }
48
49     }
50 }
```

3 tombol klik untuk Home, Shop, dan Contact untuk membuat active content pada halaman sesuai tombol yang diklik

```
29
30
31     // menempatkan konten ke posisi paling depan
32     private void ShopBtn_Click(object sender, EventArgs e)
33     {
34         ShopContent.BringToFront();
35     }
36
37     // menempatkan konten ke posisi paling depan
38     private void ContactBtn_Click(object sender, EventArgs e)
39     {
40         ContactContent.BringToFront();
41     }
42
43     // menempatkan konten ke posisi paling depan
44     private void HomeBtn_Click(object sender, EventArgs e)
45     {
46         HomeContent.BringToFront();
47     }
48
49     }
50 }
```

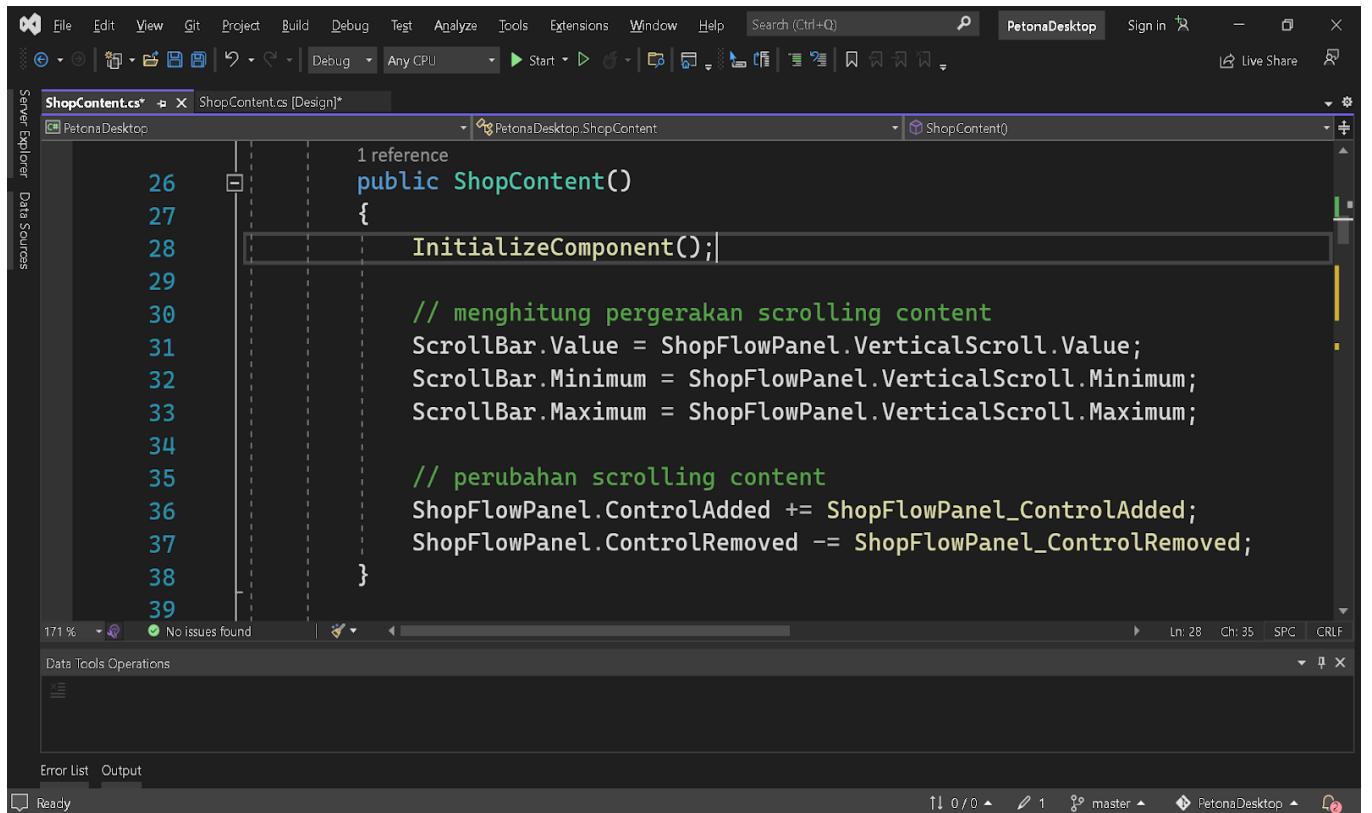
Design halaman Shop dengan membuat User Control(Windows Form) baru yang berisikan 1 Panel dengan nama ShopFlowPanel sebagai tempat dimana produk akan ditampilkan dan 1 Vertical Scroll Bar untuk scrolling produk ketika produk yang dimiliki banyak



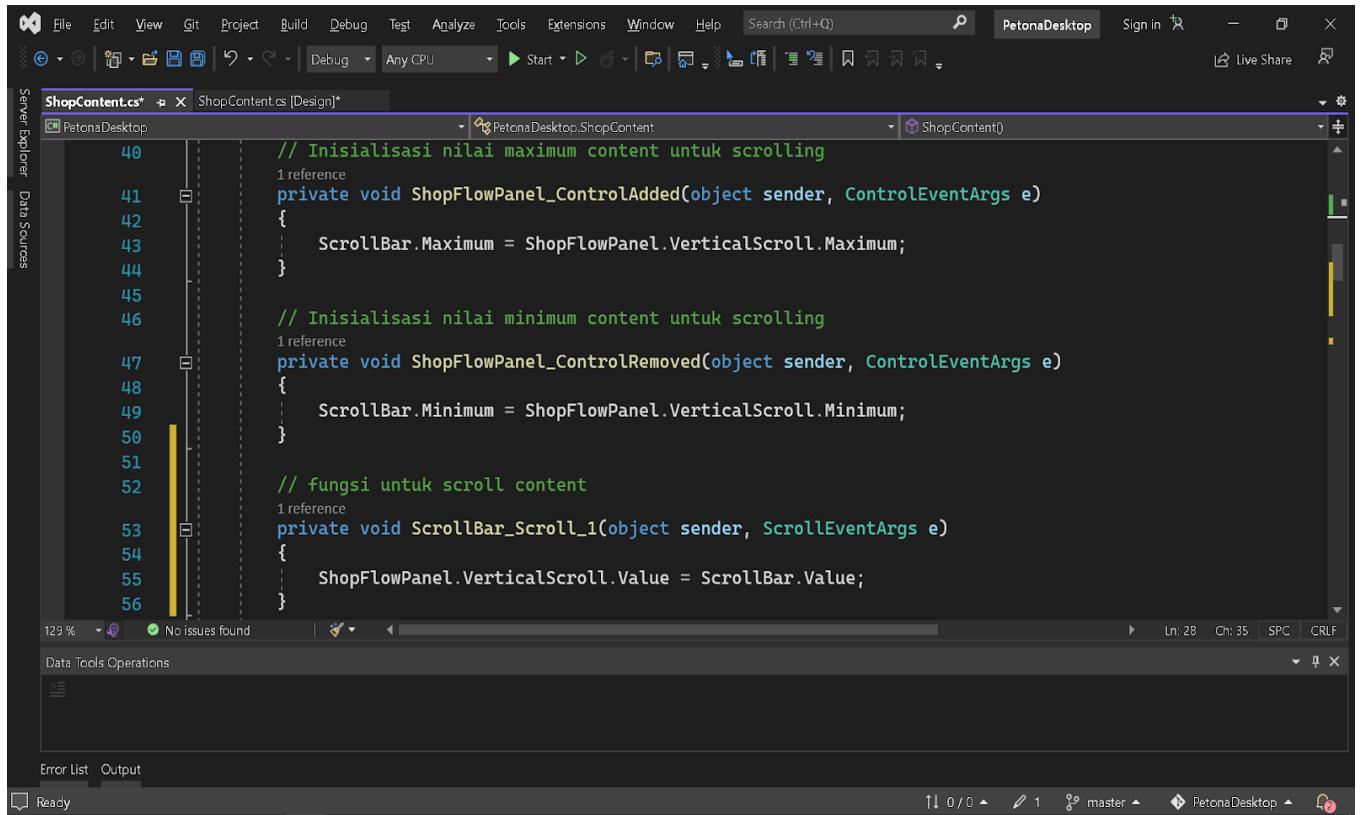
Inisialisasi koneksi dan variabel koneksi ke database mysql

```
1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel;
4  using System.Data;
5  using System.Drawing;
6  using System.Linq;
7  using System.Net.Http;
8  using System.Net;
9  using System.Text;
10 using System.Threading.Tasks;
11 using System.Windows.Forms;
12 using MySQL.Data;
13 using MySQL.Data.MySqlClient;
14 using Newtonsoft.Json;
15
16 namespace PetonaDesktop
17 {
18     public partial class ShopContent : UserControl
19     {
20         // inisialisasi koneksi ke database
21         string connStr = "server=mysql6.freesqldatabase.com;port=3306;database=mysql6492184;uid=mysql6492184;pwd=MxvYZwjszD;";
22
23         // inisialisasi variabel koneksi mysql
24         MySqlConnection conn;
```

Membuat fungsi untuk scrolling halaman produk dimana value dari scrollbar akan berubah sesuai dengan penambahan control baru pada ShopFlowPanel

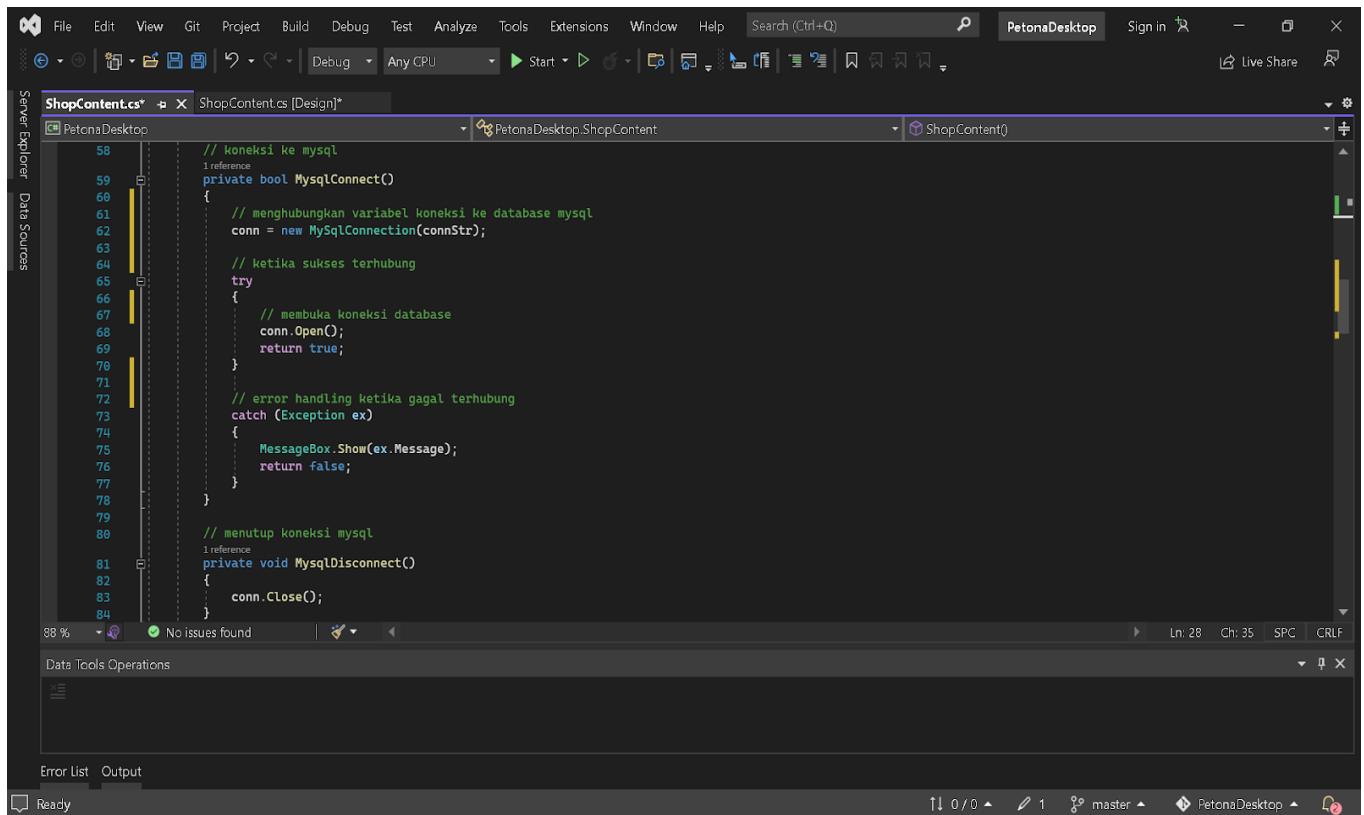


```
ShopContent.cs*  ShopContent.cs [Design]* PetonaDesktop Sign in Live Share
File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) PetonaDesktop
Any CPU Start | 
ShopContent.cs  PetonaDesktop.ShopContent  ShopContent()
1 reference
26     public ShopContent()
27     {
28         InitializeComponent();
29
30         // menghitung pergerakan scrolling content
31         ScrollBar.Value = ShopFlowPanel.VerticalScroll.Value;
32         ScrollBar.Minimum = ShopFlowPanel.VerticalScroll.Minimum;
33         ScrollBar.Maximum = ShopFlowPanel.VerticalScroll.Maximum;
34
35         // perubahan scrolling content
36         ShopFlowPanel.ControlAdded += ShopFlowPanel_ControlAdded;
37         ShopFlowPanel.ControlRemoved -= ShopFlowPanel_ControlRemoved;
38
39     }
171 % No issues found Data Tools Operations Error List Output Ready
Ln: 28 Ch: 35 SPC CRLF
```



```
ShopContent.cs*  ShopContent.cs [Design]* PetonaDesktop Sign in Live Share
File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) PetonaDesktop
Any CPU Start | 
ShopContent.cs  PetonaDesktop.ShopContent  ShopContent()
1 reference
40     // Inisialisasi nilai maximum content untuk scrolling
41     private void ShopFlowPanel_ControlAdded(object sender, ControlEventArgs e)
42     {
43         ScrollBar.Maximum = ShopFlowPanel.VerticalScroll.Maximum;
44     }
45
46     // Inisialisasi nilai minimum content untuk scrolling
47     private void ShopFlowPanel_ControlRemoved(object sender, ControlEventArgs e)
48     {
49         ScrollBar.Minimum = ShopFlowPanel.VerticalScroll.Minimum;
50     }
51
52     // fungsi untuk scroll content
53     private void ScrollBar_Scroll_1(object sender, ScrollEventArgs e)
54     {
55         ShopFlowPanel.VerticalScroll.Value = ScrollBar.Value;
56     }
129 % No issues found Data Tools Operations Error List Output Ready
Ln: 28 Ch: 35 SPC CRLF
```

Menghubungkan, membuka dan menutup koneksi ke database MySQL



The screenshot shows the Visual Studio IDE interface with the following details:

- Menu Bar:** File, Edit, View, Git, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help.
- Search Bar:** Search (Ctrl+Q).
- Toolbar:** Includes icons for Undo, Redo, Save, and various project management tools.
- Project Explorer:** Shows "PetonaDesktop" as the active project.
- Code Editor:** Displays the file "ShopContent.cs" with the following code:

```
// koneksi ke mysql
private bool MysqlConnect()
{
    // menghubungkan variabel koneksi ke database mysql
    conn = new MySqlConnection(connStr);

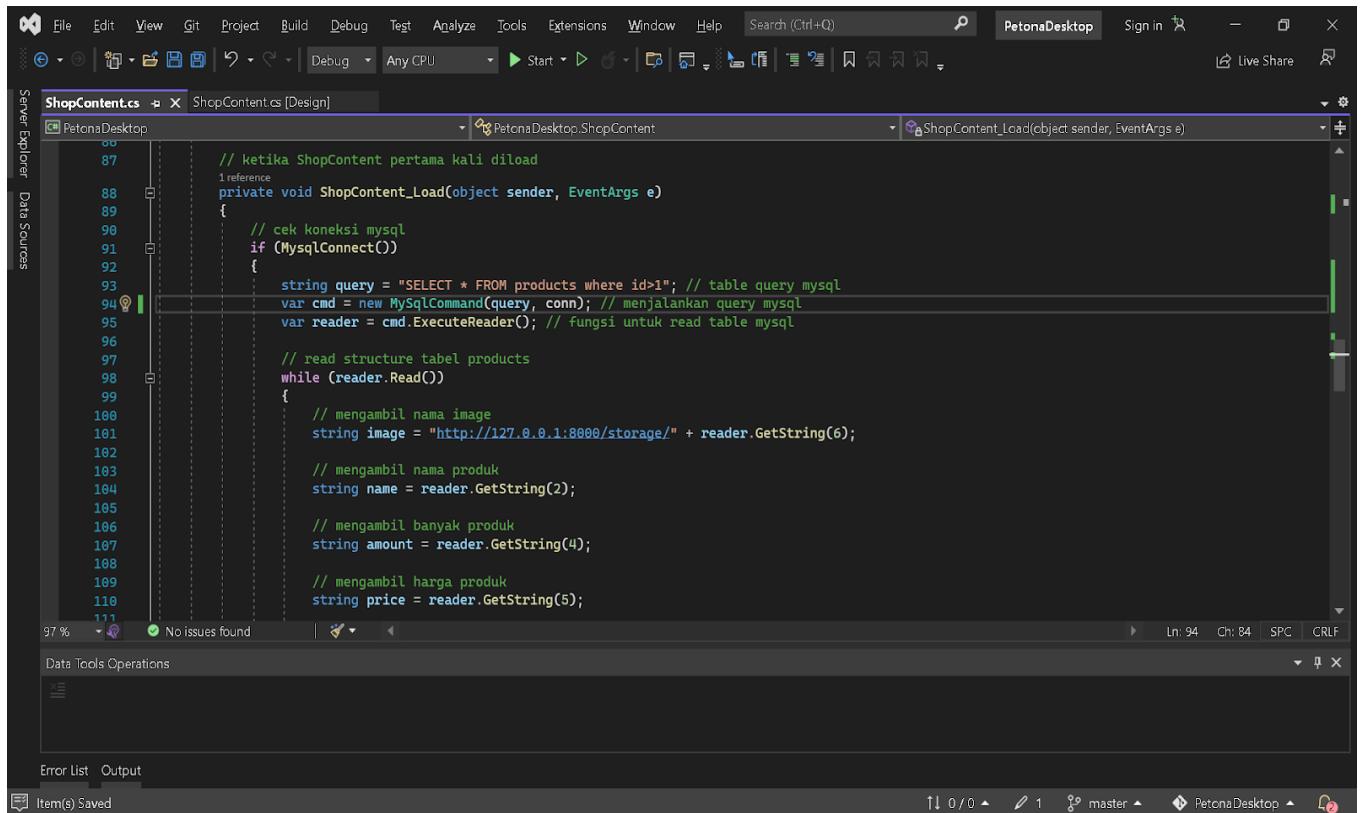
    // ketika sukses terhubung
    try
    {
        // membuka koneksi database
        conn.Open();
        return true;
    }

    // error handling ketika gagal terhubung
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
        return false;
    }
}

// menutup koneksi mysql
private void MysqlDisconnect()
{
    conn.Close();
}
```

The code handles connecting to a MySQL database using a connection string and catching exceptions if the connection fails. It also includes a method to disconnect from the database.

Ketika halaman Shop pertama kali di load dan koneksi ke database sukses maka variabel reader akan membaca structure tabel pada database untuk diambil data nama image, nama produk, banyak produk, dan harga produk. Lalu hasil reader tadi akan disimpan ke dalam masing-masing variable . Untuk nama image diambil dari api yang sudah diaktifkan dengan url <http://127.0.0.1:8000/storage/> ditambah dengan nama image yang ada di dalam database.



The screenshot shows the Visual Studio IDE interface with the following details:

- Title Bar:** PetonaDesktop
- Menu Bar:** File, Edit, View, Git, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help, Search (Ctrl+Q)
- Toolbar:** Standard toolbar icons.
- Code Editor:** The main window displays the `ShopContent.cs` file under the `PetonaDesktop` project. The code is written in C# and contains logic to connect to MySQL, execute a query, and read data from a table named `products`. It then processes each row to extract the image URL, product name, quantity, and price, and appends the image URL to the product name.
- Servers Explorer:** Shows the connection to the MySQL database.
- Data Sources:** Shows the `products` table.
- Toolbars:** Data Tools Operations, Error List, Output.
- Status Bar:** Shows the current file is `ShopContent.cs`, the branch is master, and the commit count is 1.

```
// ketika ShopContent pertama kali diload
private void ShopContent_Load(object sender, EventArgs e)
{
    // cek koneksi mysql
    if (MysqlConnect())
    {
        string query = "SELECT * FROM products where id>1"; // table query mysql
        var cmd = new MySqlCommand(query, conn); // menjalankan query mysql
        var reader = cmd.ExecuteReader(); // fungsi untuk read table mysql

        // read structure tabel products
        while (reader.Read())
        {
            // mengambil nama image
            string image = "http://127.0.0.1:8000/storage/" + reader.GetString(6);

            // mengambil nama produk
            string name = reader.GetString(2);

            // mengambil banyak produk
            string amount = reader.GetString(4);

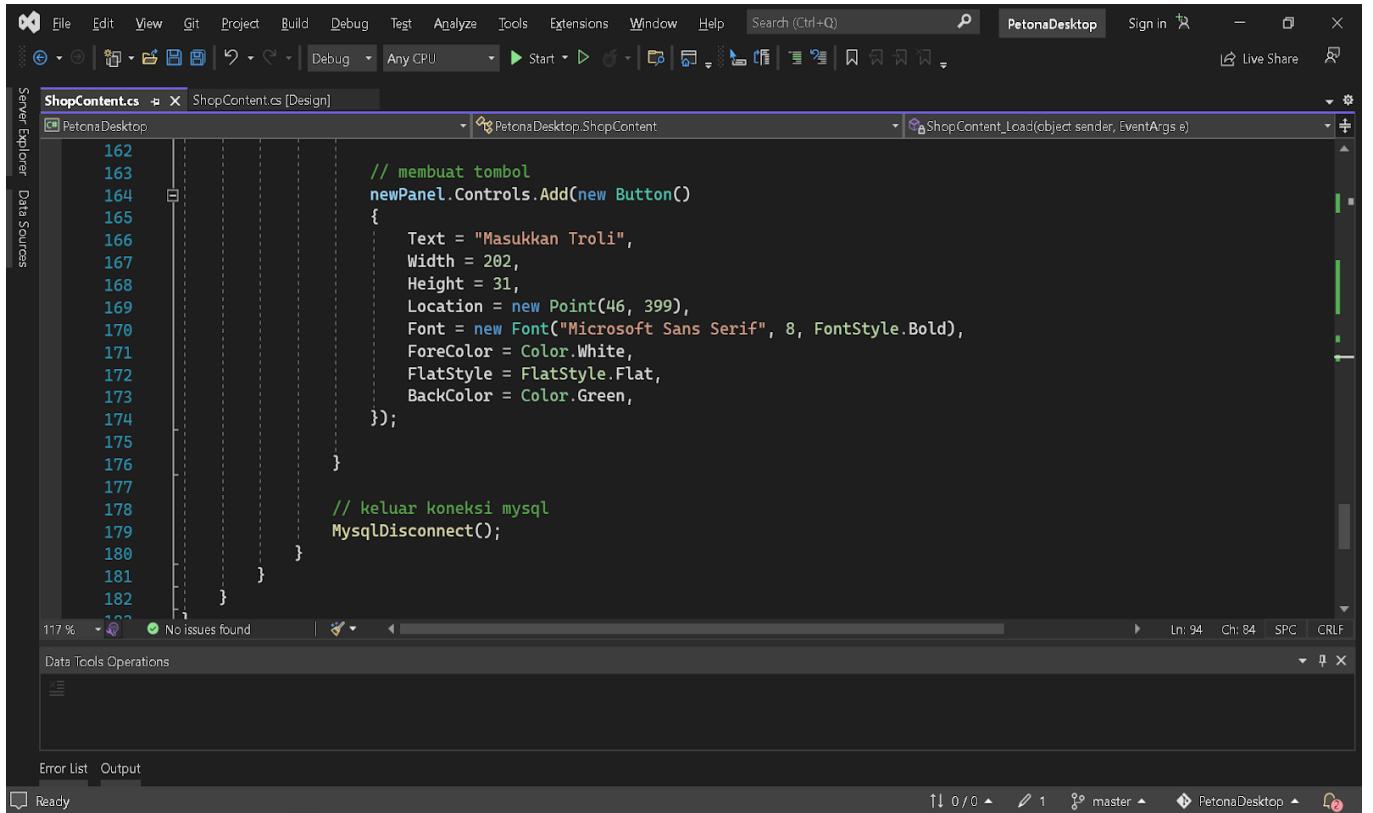
            // mengambil harga produk
            string price = reader.GetString(5);
        }
    }
}
```

Membuat komponen produk dengan panel yang berisikan 1 picturebox untuk gambar produk dan 3 label untuk nama, banyak produk, dan harga produk

```
112 // membuat komponen product
113 Panel newPanel = new Panel()
114 {
115     Width = 300,
116     Height = 455,
117     BackColor = Color.White,
118 };
119
120 // menempatkan panel
121 ShopFlowPanel.Controls.Add(newPanel);
122
123 // membuat gambar produk
124 newPanel.Controls.Add(new PictureBox())
125 {
126     ImageLocation = image,
127     Width = 300,
128     Height = 300,
129    SizeMode = PictureBoxSizeMode.Zoom,
130     Size = new Size(300,200)
131 };
132
```

```
134 // membuat label nama produk
135 newPanel.Controls.Add(new Label())
136 {
137     Text = name,
138     Location = new Point(81, 216),
139     Size = new Size(158, 29),
140     Font = new Font("Microsoft Sans Serif", 12)
141 };
142
143 // membuat label pieces
144 newPanel.Controls.Add(new Label())
145 {
146     Text = amount + " pieces",
147     Location = new Point(103, 245),
148     Size = new Size(77, 20),
149     Font = new Font("Microsoft Sans Serif", 8)
150 };
151
152 // membuat label harga
153 newPanel.Controls.Add(new Label())
154 {
155     Text = "Rp. " + price + " /kg",
156     Location = new Point(67, 362),
157     Width = 166,
158     Height = 29,
159     Font = new Font("Microsoft Sans Serif", 12, FontStyle.Bold)
160 };
161
```

Membuat tombol didalam panel untuk produk dimana tombol ini akan digunakan untuk memasukkan produk ke dalam troli belanja lalu menutup koneksi ke database dengan fungsi MysqlDisconnect()



The screenshot shows the Visual Studio IDE interface with the code editor open. The file being edited is `ShopContent.cs`. The code is written in C# and contains logic for creating a button and disconnecting from a MySQL database. The code is as follows:

```
162
163
164     // membuat tombol
165     newPanel.Controls.Add(new Button()
166     {
167         Text = "Masukkan Troli",
168         Width = 202,
169         Height = 31,
170         Location = new Point(46, 399),
171         Font = new Font("Microsoft Sans Serif", 8, FontStyle.Bold),
172         ForeColor = Color.White,
173         FlatStyle = FlatStyle.Flat,
174         BackColor = Color.Green,
175     });
176
177
178     // keluar koneksi mysql
179     MysqlDisconnect();
180 }
181
182 }
```

The code editor has syntax highlighting for C# and includes line numbers. Below the code editor, there are several toolbars and panels, including the Data Tools Operations panel and the Error List panel.

Hasil dari Halaman Shop



The screenshot shows the Petona desktop application window. The left sidebar is green and contains navigation links for HOME, SHOP, and CONTACT. The main content area has a title "Tampilkan Semua Produk". Below the title is a grid of 10 product cards, each featuring an image of a fruit, its name, quantity, and price. The products are arranged in two rows of five. The first row includes Apel, Pisang, Buah Naga, Anggur, and Lemon. The second row includes Leci, Mangga, Melon, Jeruk, and Pepaya.

Produk	Detail	Harga
Apel	50 pieces	Rp. 10000
Pisang	20 pieces	Rp. 15000
Buah Naga	10 pieces	Rp. 5000 /kg
Anggur	12 pieces	Rp. 16000
Lemon	8 pieces	Rp. 25000
Leci	5 pieces	Rp. 35000
Mangga	34 pieces	Rp. 14000
Melon	30 pieces	Rp. 24000
Jeruk	16 pieces	Rp. 7000 /kg
Pepaya	10 pieces	Rp. 6000 /kg