

LAPORAN PROYEK AKHIR SEMESTER GASAL
SMK NEGERI 1 KANDEMAN
TAHUN PELAJARAN 2025/2026



Disusun oleh:

Nama : Shafa Aisyah Karim
Kelas : X PPLG 2
NIS : 258774
Judul Proyek : Drinks Vending Machine

PENGEMBANGAN PERANGKAT LUNAK DAN GIM
SMK NEGERI 1 KANDEMAN
TAHUN PELAJARAN 2025/2026

DESKRIPSI PROYEK

Proyek ini merupakan aplikasi penjualan minuman kaleng berbasis Windows Forms (C#) yang menampilkan katalog produk menyerupai antarmuka *vending machine*. Secara keseluruhan, proyek ini merupakan simulasi sederhana dari sistem pembelian otomatis.

A. Fitur Utama Proyek

1. Pemilihan Produk

Pengguna dapat memilih minuman dengan menekan tombol atau klik tombol '*Buy*'.

Setiap produk sudah ditampilkan secara visual sehingga pengguna dapat langsung mengenali jenis minuman tanpa membaca nama produk.

2. Menampilkan Harga Produk

Setiap minuman dilengkapi dengan label harga yang jelas agar pengguna dapat mengetahui berapa biaya yang harus dikeluarkan sebelum membeli.

3. Metode Pembayaran

Fitur ini berfungsi untuk memproses transaksi setelah pengguna memilih produk.

Pada *vending machine* ini, pembayaran dapat menggunakan *e-wallet* dan *mobile banking*.

B. Tujuan Proyek

Tujuan proyek vending machine ini adalah untuk membuat aplikasi yang mampu menampilkan berbagai pilihan minuman, menyediakan proses pemilihan dan pembayaran yang sederhana. Proyek ini juga bertujuan memberikan pengalaman penggunaan yang mudah, cepat, dan menyerupai *vending machine*.

RINGKASAN TEORI SINGKAT TENTANG C# DAN MATERI YANG DIGUNAKAN

1. Konsep Dasar C#

C# adalah bahasa pemrograman **Object-Oriented Programming (OOP)**. Dalam OOP, program disusun dalam bentuk *class* dan *object*.

- a. Class adalah cetak biru yang berisi data dan fungsi.
- b. Object adalah instance dari *class* tersebut.

2. Constructor

Constructor adalah metode khusus yang dijalankan saat sebuah object dibuat.

Contoh:

```
public Form3(string PaymentMethod, string drinkName, string price)
{
    InitializeComponent();
    this.PaymentMethod = PaymentMethod;
    this.drinkName = drinkName;
    this.price = price;
}
```

Constructor digunakan untuk menerima dan menyimpan data yang dikirim dari form lain, seperti nama produk, harga, dan metode pembayaran.

3. Variabel dan Tipe Data

Proyek menggunakan beberapa tipe data dasar:

- a. string → menyimpan teks (nama minuman, harga, metode pembayaran).
- b. private variable → menyimpan data agar dapat diakses oleh seluruh bagian form.

Contoh:

```
private string PaymentMethod;
private string drinkName;
private string price;
```

4. Event-Driven Programming

C# Windows Forms bekerja dengan konsep *event-driven*, yaitu kode dijalankan ketika suatu peristiwa terjadi.

Beberapa event yang digunakan:

- a. Button.Click → saat tombol ditekan
- b. Form.Load → saat form ditampilkan
- c. ComboBox.SelectedIndexChanged → saat pilihan *ComboBox* berubah

Contoh:

```
1 reference
private void btnBuy_Click(object sender, EventArgs e)
{
```

5. Windows Forms Controls

Proyek memanfaatkan beberapa komponen (*controls*) dari *Windows Forms*:

- a. Button → menjalankan perintah (*Buy, Cancel, Confirm Payment*)
- b. Label → menampilkan informasi (nama produk, harga, metode pembayaran)
- c. ComboBox → memilih metode pembayaran
- d. Form → halaman utama tempat antarmuka ditampilkan

Setiap control memiliki properti seperti `.Text`, `.SelectedItem`, `.Show()`, `.Hide()`.

6. Pengiriman Data Antar Form

Komunikasi antar form dilakukan menggunakan parameter *constructor*.

Form1 mengirim data ke Form2, dan Form2 mengirim data ke Form3.

Contoh:

```
Form2 form2 = new Form2(drinkName, price);
form2.Show();
```

Teknik ini umum digunakan untuk meneruskan informasi antarmuka dalam aplikasi *Windows Form*.

7. Struktur Keputusan (IF-ELSE)

Struktur kondisi digunakan untuk memeriksa logika tertentu misalnya memastikan metode pembayaran telah dipilih.

Contoh:

```
if (comboBoxEWallet.SelectedItem != null)
    paymentMethod = comboBoxEWallet.SelectedItem.ToString();
else if (comboBoxMobileBanking.SelectedItem != null)
    paymentMethod = comboBoxMobileBanking.SelectedItem.ToString();
else
{
    MessageBox.Show("Lu mau bayar pake apa?");
    return;
}
```

8. String Interpolation

String interpolation digunakan untuk menampilkan informasi secara dinamis pada label.

Contoh:

```
labeldrinkName.Text = $"Product Name : {drinkName}";
```

9. Navigasi Antar Form

Dalam aplikasi *Windows Forms*, navigasi dilakukan menggunakan metode:

- a. `.Show()` → menampilkan form baru
- b. `.Hide()` → menyembunyikan form saat ini

10. Pembatasan Akses dan Enkapsulasi

Variabel-varibel seperti ‘`private string price;`’ menggunakan modifier private, yang berarti hanya dapat diakses dari dalam *class* yang sama. Ini merupakan prinsip *enkapsulasi* dalam OOP untuk menjaga keamanan data.

IMPLEMENTASI PROGRAM

A. Potongan Kode Program Utama

1. Form1.cs – Main Application

```
namespace PROYEEEEEEKKKKKKKKKKKKKKKKK
{
    8 references
    public partial class Form1 : Form
    {
        3 references
        public Form1()
        {
            InitializeComponent();
        }

        0 references
        private void Form1_Load(object sender, EventArgs e)
        {

        }

        1 reference
        private void btnBuy1_Click(object sender, EventArgs e)
        {
            string drinkName = "Monster Zero Ultra White";
            string price = "Rp 35.000";
            Form2 form2 = new Form2(drinkName, price);
            form2.Show();
            this.Hide();
        }

        1 reference
        private void btnBuy2_Click(object sender, EventArgs e)
        {
            string drinkName = "Java Monster Loca Moca";
            string price = "Rp 40.000";
            Form2 form2 = new Form2(drinkName, price);
            form2.Show();
            this.Hide();
        }

        1 reference
        private void btnBuy3_Click(object sender, EventArgs e)
        {
            string drinkName = "Java Monster Salted Caramel";
            string price = "Rp 55.500";
            Form2 form2 = new Form2(drinkName, price);
            form2.Show();
            this.Hide();
        }
}
```

```
1 reference
private void btnBuy4_Click(object sender, EventArgs e)
{
    string drinkName = "Monster Assault";
    string price = "Rp 45.000";
    Form2 form2 = new Form2(drinkName, price);
    form2.Show();
    this.Hide();
}

1 reference
private void btnBuy5_Click(object sender, EventArgs e)
{
    string drinkName = "Monster Ultra Black";
    string price = "Rp 48.000";
    Form2 form2 = new Form2(drinkName, price);
    form2.Show();
    this.Hide();
}

1 reference
private void btnBuy6_Click(object sender, EventArgs e)
{
    string drinkName = "Monster";
    string price = "Rp 30.000";
    Form2 form2 = new Form2(drinkName, price);
    form2.Show();
    this.Hide();
}

1 reference
private void btnBuy7_Click(object sender, EventArgs e)
{
    string drinkName = "Red Bull";
    string price = "Rp 25.000";
    Form2 form2 = new Form2(drinkName, price);
    form2.Show();
    this.Hide();
}

1 reference
private void btnBuy8_Click(object sender, EventArgs e)
{
    string drinkName = "Sprite";
    string price = "Rp 5.500";
    Form2 form2 = new Form2(drinkName, price);
    form2.Show();
    this.Hide();
}
```

```
}

1 reference
private void btnBuy9_Click(object sender, EventArgs e)
{
    string drinkName = "Fanta Orange";
    string price = "Rp 6.500";
    Form2 form2 = new Form2(drinkName, price);
    form2.Show();
    this.Hide();
}

1 reference
private void btnBuy10_Click(object sender, EventArgs e)
{
    string drinkName = "Coca Cola Classic";
    string price = "Rp 8.500";
    Form2 form2 = new Form2(drinkName, price);
    form2.Show();
    this.Hide();
}

1 reference
private void btnBuy11_Click(object sender, EventArgs e)
{
    string drinkName = "Pepsi";
    string price = "Rp 28.500";
    Form2 form2 = new Form2(drinkName, price);
    form2.Show();
    this.Hide();
}

1 reference
private void btnBuy12_Click(object sender, EventArgs e)
{
    string drinkName = "Dr. Pepper";
    string price = "Rp 19.500";
    Form2 form2 = new Form2(drinkName, price);
    form2.Show();
    this.Hide();
}

1 reference
private void Form1_Load_1(object sender, EventArgs e)
{
}

}
```

2. Form2.cs – Payment Application

```
namespace PROVEEEEEEKKKKKKKKKKKKKKKKK
{
    26 references
    public partial class Form2 : Form
    {
        private string drinkName;
        private string price;
        12 references
        public Form2(string drinkName, string price)
        {
            InitializeComponent();
            this.drinkName = drinkName;
            this.price = price;
        }

        1 reference
        private void comboBoxEWallet_SelectedIndexChanged(object sender, EventArgs e)
        {
            comboBoxMobileBanking.SelectedIndex = -1;
        }
        1 reference
        private void comboBoxMobileBanking_SelectedIndexChanged(object sender, EventArgs e)
        {
            comboBoxEWallet.SelectedIndex = -1;
        }
        1 reference
        private void btnBuy_Click(object sender, EventArgs e)
        {
            string paymentMethod = "";

            // Cek metode pembayaran yang dipilih
            if (comboBoxEWallet.SelectedItem != null)
                paymentMethod = comboBoxEWallet.SelectedItem.ToString();
            else if (comboBoxMobileBanking.SelectedItem != null)
                paymentMethod = comboBoxMobileBanking.SelectedItem.ToString();
            else
            {
                MessageBox.Show("Lu mau bayar pake apa?");
                return;
            }
            Form3 form3 = new Form3(paymentMethod, drinkName, price);
            form3.Show();
            this.Hide();
        }

        1 reference
        private void btnCancel_Click(object sender, EventArgs e)
        {
            this.Hide();
            Form1 form1 = new Form1();
            form1.Show();
        }

        1 reference
        private void Form2_Load(object sender, EventArgs e)
        {
        }
    }
}
```

3. Form3.cs – Confirm Payment

```
namespace PROYEEEEEEKKKKKKKKKKKKKKKKK
{
    4 references
    public partial class Form3 : Form
    {
        private string PaymentMethod;
        private string drinkName;
        private string price;
        1 reference
        public Form3(string PaymentMethod, string drinkName, string price)
        {
            InitializeComponent();
            this.PaymentMethod = PaymentMethod;
            this.drinkName = drinkName;
            this.price = price;
        }

        1 reference
        private void Form3_Load(object sender, EventArgs e)
        {
            labelDrinkName.Text = $"Product Name : {drinkName}";
            labelPrice.Text = $"Price : {this.price}";
            labelPaymentMethod.Text = $"Payment Method : {this.PaymentMethod}";
        }

        1 reference
        private void label5_Click(object sender, EventArgs e)
        {
            this.Hide();
            Form1 form1 = new Form1();
            form1.Show();
        }

        1 reference
        private void label4_Click(object sender, EventArgs e)
        {
            MessageBox.Show("Payment Successful!");
        }
    }
}
```

B. Penjelasan Kode Program

1. Struktur Utama Form1.cs

a. Namespace

Berfungsi sebagai “wadah” untuk mengelompokkan *class* di dalam program agar rapi dan tidak bentrok dengan nama *class* lain.

b. Deklarasi Class Form1

```
public partial class Form1 : Form
```

Menyatakan bahwa Form1 adalah sebuah *Windows Forms* yang mewarisi fitur dari *class* Form bawaan .NET. Kata *partial* berarti kode Form1 dapat terbagi di beberapa file (misal Form1.cs dan Form1.Designer.cs).

c. Constructor Form1

```
public partial class Form1 : Form
{
    3 references
    public Form1()
    {
        InitializeComponent();
    }
}
```

InitializeComponent(); memuat seluruh desai form (tombol, label, layout, dll) yang dibuat melalui Windows Form Designer.

d. Event Handler Tombol Buy

```
private void btnBuy1_Click(object sender, EventArgs e)
{
    string drinkName = "Monster Zero Ultra White";
    string price = "Rp 35.000";
    Form2 form2 = new Form2(drinkName, price);
    form2.Show();
    this.Hide();
}
```

- ❖ Membuat dua *string*:
 - drinkName = nama minuman
 - price = harga
- ❖ Mengirim kedua data tersebut ke Form2 melalui *constructor*:
new Form2(drinkname, price);
- ❖ Menampilkan Form2 (form2.Show()) dan menyembunyikan Form1 (this.Hide()).

2. Struktur Utama Form2

a. Deklarasi Variabel Private

```
private string drinkName;
private string price;
```

Berfungsi menjadi penyimpanan data yang dikirim dari Form2 ke Form3, yaitu;

- ❖ drinkName → nama produk
- ❖ price → harga produk

Variabel dibuat *private* agar hanya dapat digunakan dalam Form3.

b. Constructor Form2

```
public Form2(string drinkName, string price)
{
    InitializeComponent();
    this.drinkName = drinkName;
    this.price = price;
}
```

- ❖ Form2(...) dipanggil saat Form2 dibuat.
- ❖ Parameter drinkName dan price berasal dari Form1.
- ❖ this.drinkName = drinkName; → menyimpan data ke dalam variabel *class*.

Jadi Form2 menerima data dari Form1.

c. Event comboBoxEWallet_SelectedIndexChanged dan comboBoxMobileBanking_SelectedIndexChanged

```
private void comboBoxEWallet_SelectedIndexChanged(object sender, EventArgs e)
{
    comboBoxMobileBanking.SelectedIndex = -1;
}
1 reference
private void comboBoxMobileBanking_SelectedIndexChanged(object sender, EventArgs e)
{
    comboBoxEWallet.SelectedIndex = -1;
}
```

Jika pengguna memilih E-Wallet, maka opsi di Mobile Banking dikosongkan. Apabila pengguna memilih Mobile Banking, maka opsi di E-Wallet dikosongkan. Ini bertujuan supaya hanya satu metode pembayaran yang bisa dipilih.

d. Event Tombol Buy (btnBuy_Click)

```
private void btnBuy_Click(object sender, EventArgs e)
{
    string paymentMethod = "";
```

Membuat variabel untuk menyimpan metode pembayaran

e. Cek Metode Pembayaran

```
// Cek metode pembayaran yang dipilih
if (comboBoxEWallet.SelectedItem != null)
    paymentMethod = comboBoxEWallet.SelectedItem.ToString();
else if (comboBoxMobileBanking.SelectedItem != null)
    paymentMethod = comboBoxMobileBanking.SelectedItem.ToString();
else
{
    MessageBox.Show("Lu mau bayar pake apa?");
    return;
}
```

Jika pengguna memilih E-Wallet, simpan pilihan. Jika tidak, cek apakah pengguna memilih mobile banking lalu simpan. Jika dua-duanya kosong, tampilkan pesan *error* dan hentikan proses.

3. Struktur Utama Form3

a. Deklarasi Variabel Private

```
private string PaymentMethod;
private string drinkName;
private string price;
```

Tiga variabel ini berfungsi untuk menyimpan data yang dikirim dari form sebelumnya, yaitu:

- ❖ PaymentMethod → metode pembayaran yang dipilih pengguna.
- ❖ drinkName → nama produk yang dibeli.
- ❖ price → harga produk.

Variabel dibuat *private* sehingga hanya dapat digunakan di *class* Form3.

b. Constructor Form3

```
public Form3(string PaymentMethod, string drinkName, string price)
{
    InitializeComponent();
    this.PaymentMethod = PaymentMethod;
    this.drinkName = drinkName;
    this.price = price;
}
```

Constructor ini berfungsi untuk memuat seluruh komponen visual Form3 melalui *InitializeComponent()*, menerima tiga parameter dari Form2, dan menyimpan nilai parameter ke variabel *class* menggunakan *this*. sehingga dapat digunakan saat Form3 ditampilkan.

c. Event Form3_Load

```
private void Form3_Load(object sender, EventArgs e)
{
    labelDrinkName.Text = $"Product Name : {drinkName}";
    labelPrice.Text = $"Price : {this.price}";
    labelPaymentMethod.Text = $"Payment Method : {this.PaymentMethod}";
}
```

Fungsi event yaitu:

- ❖ Menampilkan informasi pesanan pada tiga label:
 - labelDrinkName, menampilkan nama produk.
 - labelPrice, menampilkan harga.
 - labelPaymentMethod, menampilkan metode pembayaran.

- ❖ Data yang ditampilkan berasal dari variabel yang dikirim melalui constructor.

d. Event label4_Click

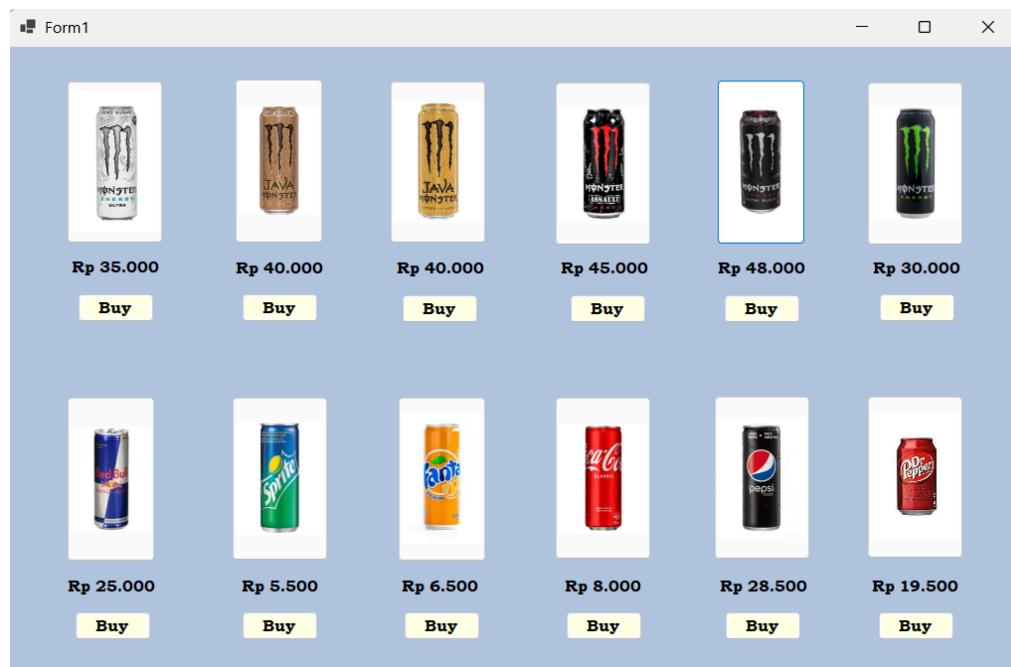
```
private void label4_Click(object sender, EventArgs e)
{
    MessageBox.Show("Yatta! Payment berhasil! >.<");
}
```

Berfungsi untuk menampilkan kotak dialog notifikasi bahwa pembayaran berhasil.

LAMPIRAN

A. Screenshot Tampilan Aplikasi

1. Form1.cs [Design]



2. Form2.cs [Design]



3. Form3.cs [Design]

