

**LAPORAN TUGAS MATA KULIAH PEMROGRAMAN VISUAL
PET SHOP MANAGEMENT SYSTEM
USING VISUAL STUDIO**



Nama Anggota :

Angelina Kismasari NIM 20051397034

Ica Nissolekha NIM 20051397064

Karina Irna Della NIM 20051397030

**PROGRAM STUDI MANAJEMEN INFORMATIKA
FAKULTAS VOKASI
UNIVERSITAS NEGERI SURABAYA**

2022

BAB I

1.1 Pengertian Visual Studio

Visual studio merupakan sebuah perangkat lunak lengkap (suite) yang dapat digunakan untuk menerapkan pengembangan aplikasi, baik itu aplikasi usaha dagang/jasa, aplikasi personal, ataupun komponen aplikasinya, dalam bangun aplikasi console, aplikasi Windows, ataupun aplikasi Web.

1.2 Pengertian Pet Shop Management

Pet Shop merupakan merupakan tempat yang menjual peralatan dan perlengkapan untuk hewan peliharaan. Management merupakan proses perencanaan, pengorganisasian, pengarahan, dan pengawasan usaha. Oleh karena itu, Pet Shop Management dapat diartikan sebagai perencanaan, pengorganisasian, pengarahan, dan pengawasan usaha peralatan dan perlengkapan kebutuhan hewan peliharaan.

1.3 Penjelasan Aplikasi

Dalam project akhir kali ini, kami membuat aplikasi berjudul **“Pet Shop Management System”**. Aplikasi ini menggunakan metode CRUD (Create Read Update Delete) sebagai manajemen data di setiap formnya. Selain itu fungsi dari aplikasi ini dapat digunakan untuk mengelola usaha Pet Shop atau peralatan dan perlengkapan hewan peliharaan. Adapun produk yang kami pasarkan adalah makanan hewan peliharaan, peralatan hewan peliharaan, serta obat-obatan untuk hewan peliharaan. Software yang digunakan untuk mengembangkan aplikasi ini adalah Microsoft Visual Studio 2022.

BAB II

2.1 Fitur-fitur dalam Aplikasi

1) Login dan Logout.

Fitur ini digunakan sebagai pengatur proses identifikasi yang mewajibkan karyawan untuk memasukkan username dan password. Kedua hal tersebut adalah serangkaian karakter untuk mengautentikasi karyawan.

2) Dashboard Home.

Di dalam dashboard menampilkan indikasi utama yang menunjukkan keseluruhan sistem secara sekilas. Dashboard berisi menu-menu form, profil user, judul aplikasi, grafik penjualan, dan total pemasukan yang didapat.

3) Product Form.

Form ini digunakan untuk memasukkan data produk yang telah terjual dengan memasukkan merek produk, kategori produk, jumlah produk yang dibeli, dan total harga. Nantinya data yang telah dimasukkan akan ditampilkan melalui Data Grid View berupa tabel. Karyawan juga dapat melakukan pengubahan dan penghapusan data melalui form ini.

4) Employees Form.

Fitur ini digunakan untuk mendata para karyawan pet shop serta memberinya hak akses untuk login di aplikasi. Form ini berisi nama karyawan, no telp karyawan, alamat karyawan, serta password. Jika seorang karyawan belum memiliki akses untuk login di aplikasi maka karyawan tersebut harus melapor ke manajer untuk didata dan diberikan akses berupa password.

5) Customers Form.

Form ini digunakan untuk mendata para pengunjung yang datang. Nantinya data dari form ini akan digunakan untuk melengkapi form billing berupa struk pembayaran. Di dalamnya terdapat nama pengunjung, no telp pengunjung, dan alamat pengunjung. Namun untuk pengisian no telp dan alamat tidak diwajibkan.

6) Billings Form.

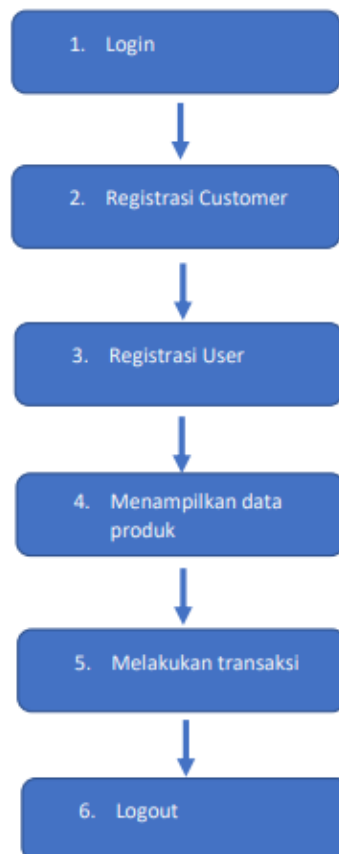
Fitur billing berisi ID dan nama pengunjung serta data produk yang dibeli. Nantinya akan muncul struk pembayaran yang berisi data lengkap pembelian hingga total harga yang harus dibayarkan..

2.2 Dokumen Teknis Petunjuk Aplikasi

A. Penjelasan Umum

Panduan ini merupakan dokumen teknis yang berisi tahapan dan informasi dari aplikasi Pet Shop Management yang bertujuan agar pengguna dapat memahami fungsi dari aplikasi tersebut. Berikut adalah Alur pokok penggunaan Aplikasi Pet Shop Management.

ALUR PENGGUNAAN APLIKASI PET SHOP MANAGEMENT



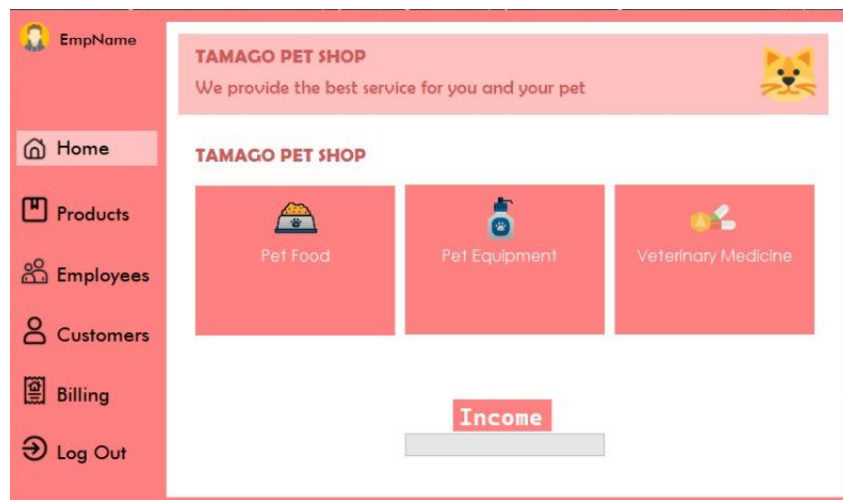
B. Dashboard dan Menu

- 1) Pada tampilan login terdapat kolom username dan password. Masukkan username dan password yang sesuai, kemudian pilih tombol **LOGIN**.



The login form is titled "LOGIN" in bold red text. Below the title, it says "Silahkan login terlebih dahulu untuk melanjutkan" in a smaller red font. There are two input fields: "Username" with a person icon and "Password" with a lock icon. Both fields are white with red borders. Below the password field is a red "LOGIN" button and a red "Reset" link.

- 2) Setelah berhasil login, akan masuk ke tampilan Dashboard yang berisi profil user, menu, nama toko beserta slogannya, grafik penjualan untuk setiap kategori, dan total pemasukan yang didapat.



The dashboard is titled "TAMAGO PET SHOP" in bold red text. Below the title, it says "We provide the best service for you and your pet" in a smaller red font. There is a cat icon in the top right corner. On the left side, there is a red sidebar with a user profile icon and the text "EmpName". Below the sidebar, there are six menu items: "Home", "Products", "Employees", "Customers", "Billing", and "Log Out". The main content area has three red boxes labeled "Pet Food", "Pet Equipment", and "Veterinary Medicine". Below these boxes is a red "Income" button and a grey bar.

- 3) Terdapat 6 menu dan tombol logout di Dashboard. Menu-menu tersebut diantaranya Home, Product, Employees, Customers, dan Billings.
- 4) Masuk ke dalam form produk untuk mendata produk yang dibeli oleh pengunjung dengan memasukkan merek produk, kategori produk, jumlah produk yang dibeli, dan total harga. Jika selesai pilih **SAVE**. Namun jika ingin melakukan perubahan maka klik baris data yang ingin diubah lalu ubah sesuai keinginan.

Jika selesai pilih **EDIT**. Jika ingin menghapus data maka pilih baris data yang ingin dihapus lalu pilih **DELETE**.

- 5) Masuk ke dalam form customers untuk mendata pengunjung yang berisi nama pengunjung, no telp pengunjung, dan alamat pengunjung. Apabila pengunjung tidak berkenan untuk memberikan no telp dan alamat maka bisa ditulis dengan “-”. Fungsi dari no telp dan alamat adalah untuk layanan delivery. Jika selesai klik tombol **SAVE**. Namun jika ingin melakukan pengubahan maka klik baris data yang ingin diubah lalu ubah sesuai keinginan. Jika selesai pilih **EDIT**. Jika ingin menghapus data maka pilih baris data yang ingin dihapus lalu pilih **DELETE**.

The screenshot shows a web application interface with a red sidebar on the left containing navigation links: Home, Products, Employees, Customers (highlighted), Billing, and Log Out. The main content area is titled 'CUSTOMERS' and features a form with input fields for Name, Phone, and Address, along with Save, Edit, and Delete buttons. Below the form is a 'CUSTOMERS LIST' table with columns CustID, CustName, CustAdd, and CustPhone. The first row of the table is highlighted in blue.

CustID	CustName	CustAdd	CustPhone
1	Nanami	Bumi	08815075942

- 6) Masuk ke form billings untuk melanjutkan pembayaran. Di dalamnya terdapat dua kolom yakni CUSTOMER dan PRODUCT.
- Pada kolom CUSTOMER masukkan ID customer dan namanya. Jika selesai klik tombol **SAVE**. Namun jika ingin melakukan pengubahan maka klik baris data yang ingin diubah lalu ubah sesuai keinginan. Jika selesai pilih **EDIT**.
 - Pada kolom PRODUCT masukkan merek produk, jumlah produk yang dibeli, dan harga produk. Jika selesai klik tombol **SAVE**. Namun jika ingin mengulang pilih **RESET**.

Nantinya akan muncul struk pembayaran yang berisi produk yang dibeli beserta total harganya.

- 7) Jika ingin memasukkan data karyawan baru dan memberinya akses berupa password maka masuk ke form employee. Masukkan nama karyawan, no telp karyawan, alamat karyawan, serta password. Jika selesai klik tombol **SAVE**. Namun jika ingin melakukan pengubahan maka klik baris data yang ingin diubah lalu ubah sesuai keinginan. Jika selesai pilih **EDIT**. Jika ingin menghapus data maka pilih baris data yang ingin dihapus lalu pilih **DELETE**.

EMPLOYEES

Name Phone Password Address

Date of Birth Kamis, 26 Mei 2022

Save Edit Delete

EMPLOYEES LIST

EmpID	EmpName	EmpAdd	EmpDOB	EmpPhone	EmpPass
1	Lina	Isakai	21/07/2002	08815075943	12345
2	Angel	Bumi	20/07/2002	08123456789	00000
*					

- 8) Jika ingin keluar maka klik tombol logout untuk menutup halaman utama aplikasi.

BAB III

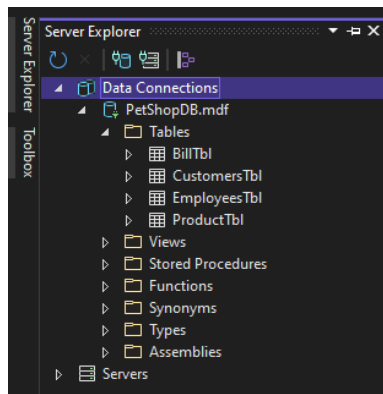
3.1 Kontribusi Setiap Anggota Kelompok

1) Anggelina Kismasari

a) Uraian Pengerjaan

1. Data Connections

- Membuat database melalui server explorer menggunakan SQL Client Server yang diberi nama “PetShopDB.mdf”.



- Membuat beberapa tabel (Billing, Customer, Employee, Product).

	Bld	BDate	CustID	CustName	EmpName	Amt
	NULL	NULL	NULL	NULL	NULL	NULL

	CustID	CustName	CustAdd	CustPhone
	1	Nanami	Bumi	08815075942
	NULL	NULL	NULL	NULL

	EmpID	EmpName	EmpAdd	EmpDOB	EmpPhone	EmpPass
	1	Lina	Isekai	21/07/2002	08815075943	12345
	2	Angel	Bumi	20/07/2002	08123456789	00000
	NULL	NULL	NULL	NULL	NULL	NULL

	ProId	ProName	ProCat	ProQty	ProPrice
	1	Royal Canin	Pet Food	1	45000
	NULL	NULL	NULL	NULL	NULL

- Mengkoneksikan setiap form yang membutuhkan koneksi database.

2. Product Form

- Mengatur ukuran form sebesar 850 x 500 dengan start position (center screen) dan form border style (none). Hal ini agar saat form dijalankan maka dia akan muncul di tengah layar tanpa menggunakan form border style.
- Menggunakan beberapa label yang berisi tulisan dan icon untuk membuat tampilan menu di samping kiri form dan profil user di pojok label kiri atas.
- Membuat label yang dinamai nama produk, kategori produk, jumlah produk, dan harga serta icon kucing di pojok label kanan atas.
- Memasukkan tiga text box dan satu combo box yang berisi kategori penjualan (makanan, peralatan, obat-obatan).
- Memasukkan tiga button untuk membuat save button, edit button, dan delete button.
- Membuat product list menggunakan satu label dan data grid view untuk menampilkan data yang dimasukkan.
- Mengkoneksikan form dengan database dan melakukan koding untuk setiap button yang ada untuk dihubungkan pada data grid view.

3. Billing Form

- Memasukkan beberapa label dan icon untuk membuat tampilan menu yang sama seperti form produk.
- Membagi halaman menggunakan 2 label dan 2 data grid view yang menampilkan data dari setiap tabel yang berbeda (input data customer dan input data produk).
- Pada halaman customer menggunakan 1 text box, 2 button, dan 1 combo box yang berisi ID customer. Data dari combo box tersebut didapatkan dari form customer.

- Pada halaman product menggunakan 3 text box dan 2 button untuk memasukkan data produk yang akan ditampilkan pada struk pembayaran.
- Pada halaman billing belum dilakukan pengkodean, masih hanya berupa desain UI saja.

b) Screenshot Koding

1. Pengkoneksian Database di Form Employee, Product, Customer

```
SqlConnection Con = new SqlConnection(@"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\ASUS\Documents\Employee.mdf;User Id=sa;Password=1234567890;");
private void DisplayEmployees()
{
    Con.Open();
    string Query = "Select * from EmployeesTbl";
    SqlDataAdapter sda = new SqlDataAdapter(Query, Con);
    SqlCommandBuilder Builder = new SqlCommandBuilder(sda);
    var ds = new DataSet();
    sda.Fill(ds);
    EmployeesDGV.DataSource = ds.Tables[0];
    Con.Close();
}
```

```
SqlConnection Con = new SqlConnection(@"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\ASUS\Documents\Product.mdf;User Id=sa;Password=1234567890;");
private void DisplayProducts()
{
    Con.Open();
    string Query = "Select * from ProductTbl";
    SqlDataAdapter sda = new SqlDataAdapter(Query, Con);
    SqlCommandBuilder Builder = new SqlCommandBuilder(sda);
    var ds = new DataSet();
    sda.Fill(ds);
    ProductsDGV.DataSource = ds.Tables[0];
    Con.Close();
}
```

```
SqlConnection Con = new SqlConnection(@"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\ASUS\Documents\Customer.mdf;User Id=sa;Password=1234567890;");
private void DisplayEmployees()
{
    Con.Open();
    string Query = "Select * from EmployeesTbl";
    SqlDataAdapter sda = new SqlDataAdapter(Query, Con);
    SqlCommandBuilder Builder = new SqlCommandBuilder(sda);
    var ds = new DataSet();
    sda.Fill(ds);
    EmployeesDGV.DataSource = ds.Tables[0];
    Con.Close();
}
```

2. Koding Form Product

```
12 namespace PetShop
13 {
14     5 references
15     public partial class Products : Form
16     {
17         1 reference
18         public Products()
19         {
20             InitializeComponent();
21             DisplayProducts();
22         }
23         SqlConnection Con = new SqlConnection(@"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\ASU\
24         4 references
25         private void DisplayProducts()
26         {
27             Con.Open();
28             string Query = "Select * from ProductTbl";
29             SqlDataAdapter sda = new SqlDataAdapter(Query, Con);
30             SqlCommandBuilder Builder = new SqlCommandBuilder(sda);
31             var ds = new DataSet();
32             sda.Fill(ds);
33             ProductsDGV.DataSource = ds.Tables[0];
34             Con.Close();
35         }
36         3 references
37         private void Clear()
38         {
39             PrNameTb.Text = "";
40             CategoryCb.SelectedIndex = 0;
41             QtyTb.Text = "";
42             PriceTb.Text = "";
43         }
44     }
45 }
```

```
40 1 reference
41 private void SaveBtn_Click(object sender, EventArgs e)
42 {
43     if (PrNameTb.Text == "" || CategoryCb.SelectedIndex == -1 || QtyTb.Text == "" || PriceTb.Text == "")
44     {
45         MessageBox.Show("Missing Information");
46     }
47     else
48     {
49         try
50         {
51             Con.Open();
52             SqlCommand cmd = new SqlCommand("insert into ProductTbl (ProName, ProCat, ProQty, ProPrice)
53             cmd.Parameters.AddWithValue("@PN", PrNameTb.Text);
54             cmd.Parameters.AddWithValue("@PC", CategoryCb.SelectedItem.ToString());
55             cmd.Parameters.AddWithValue("@PQ", QtyTb.Text);
56             cmd.Parameters.AddWithValue("@PP", PriceTb.Text);
57             cmd.ExecuteNonQuery();
58             MessageBox.Show("Product Added");
59             Con.Close();
60             DisplayProducts();
61             Clear();
62         }
63         catch (Exception Ex)
64         {
65             MessageBox.Show(Ex.Message);
66         }
67     }
68     int Key = 0;
69 }
```

```
69 1 reference
70 private void ProductsDGV_CellContentClick(object sender, DataGridViewCellEventArgs e)
71 {
72     PrNameTb.Text = ProductsDGV.SelectedRows[0].Cells[1].Value.ToString();
73     CategoryCb.Text = ProductsDGV.SelectedRows[0].Cells[2].Value.ToString();
74     QtyTb.Text = ProductsDGV.SelectedRows[0].Cells[3].Value.ToString();
75     PriceTb.Text = ProductsDGV.SelectedRows[0].Cells[4].Value.ToString();
76     if (PrNameTb.Text == "")
77     {
78         Key = 0;
79     }
80     else
81     {
82         Key = Convert.ToInt32(ProductsDGV.SelectedRows[0].Cells[0].Value.ToString());
83     }
84 }
```

```

85 private void EditBtn_Click(object sender, EventArgs e)
86 {
87     if (PrNameTb.Text == "" || CategoryCb.SelectedIndex == -1 || QtyTb.Text == "" || PriceTb.Text == "")
88     {
89         MessageBox.Show("Missing Information");
90     }
91     else
92     {
93         try
94         {
95             Con.Open();
96             SqlCommand cmd = new SqlCommand("Update ProductTbl set ProName=@PN, ProCat=@PC, ProQty=@PQ, Price=@PP where ProId=@PKey", Con);
97             cmd.Parameters.AddWithValue("@PN", PrNameTb.Text);
98             cmd.Parameters.AddWithValue("@PC", CategoryCb.SelectedItem.ToString());
99             cmd.Parameters.AddWithValue("@PQ", QtyTb.Text);
100             cmd.Parameters.AddWithValue("@PP", PriceTb.Text);
101             cmd.Parameters.AddWithValue("@PKey", Key);
102             cmd.ExecuteNonQuery();
103             MessageBox.Show("Product Update");
104             Con.Close();
105             DisplayProducts();
106             Clear();
107         }
108         catch (Exception Ex)
109         {
110             MessageBox.Show(Ex.Message);
111         }
112     }
113 }

```

```

115 private void DeleteBtn_Click(object sender, EventArgs e)
116 {
117     if (Key == 0)
118     {
119         MessageBox.Show("Select an product");
120     }
121     else
122     {
123         try
124         {
125             Con.Open();
126             SqlCommand cmd = new SqlCommand("delete from ProductTbl where ProId = @PKey", Con);
127             cmd.Parameters.AddWithValue("@PKey", Key);
128             cmd.ExecuteNonQuery();
129             MessageBox.Show("Product Deleted");
130             Con.Close();
131             DisplayProducts();
132             Clear();
133         }
134         catch (Exception Ex)
135         {
136             MessageBox.Show(Ex.Message);
137         }
138     }
139 }
140 }
141 }
142 }

```

2) Ica Nissolekha

a) Uraian Pengerjaan

1. Splash Form

- Terdapat 2 label untuk tulisan judul.
- Terdapat 1 label untuk icon.
- Terdapat loading bar

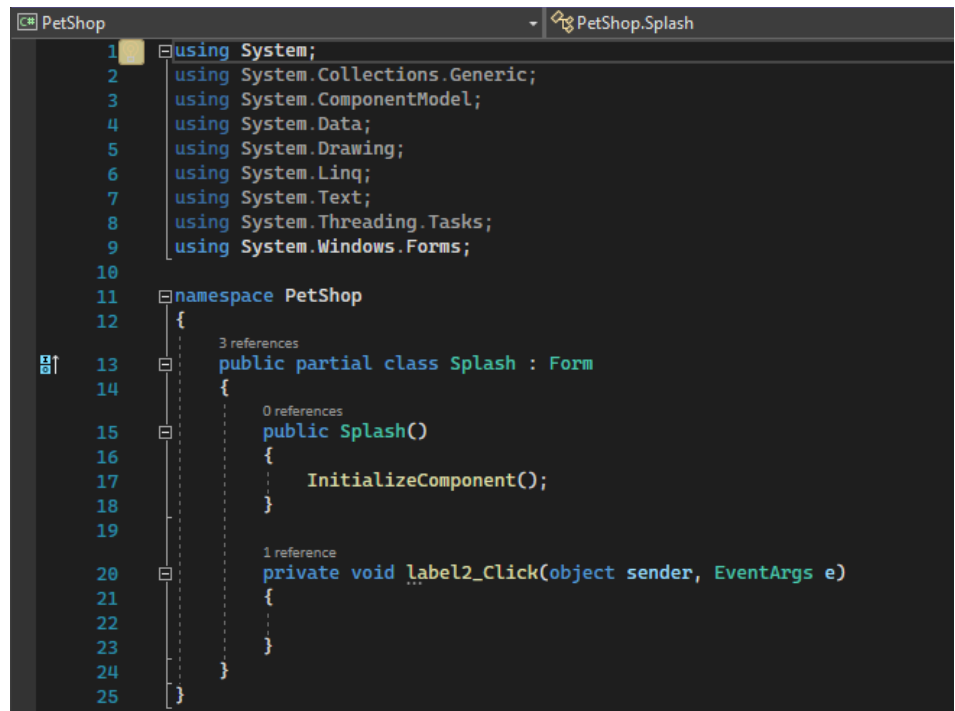
2. Customers Form

- Terdapat beberapa Text box yaitu 'Name', 'Phone', dan 'Address' untuk mengisi data kustomer.
- Button 'Save' untuk menyimpan.

- Button 'Edit' untuk mengubah data.
- Button 'Delete' untuk menghapus data.
- Data Grid View dengan label 'Customers List' untuk menampilkan data kustomer.

b) Screenshot Kodingan

1. Splash Form



```
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.Text;
8 using System.Threading.Tasks;
9 using System.Windows.Forms;
10
11 namespace PetShop
12 {
13     public partial class Splash : Form
14     {
15         public Splash()
16         {
17             InitializeComponent();
18         }
19
20         private void label2_Click(object sender, EventArgs e)
21         {
22         }
23     }
24 }
25
```

2. Customers Form

```
Customers.cs [Design] Splash.cs [Design]
PetShop PetShop.Customers Customers()

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.Text;
8 using System.Threading.Tasks;
9 using System.Windows.Forms;
10 using System.Data.SqlClient;
11
12 namespace PetShop
13 {
14     5 references
15     public partial class Customers : Form
16     {
17         1 reference
18         public Customers()
19         {
20             InitializeComponent();
21             DisplayCustomers();
22         }
23         SqlConnection Con = new SqlConnection(@"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\ASUS\Documents\PetShop
24         4 references
25         private void DisplayCustomers()
26         {
27             Con.Open();
28             string Query = "Select * from CustomersTbl";
29             SqlDataAdapter sda = new SqlDataAdapter(Query, Con);
30             SqlCommandBuilder Builder = new SqlCommandBuilder(sda);
31             var ds = new DataSet();
32             sda.Fill(ds);
33             CustomersDGV.DataSource = ds.Tables[0];
34             Con.Close();
35         }
36     }
37 }
```

```
Customers.cs [Design] Splash.cs [Design]
PetShop PetShop.Customers Customers()

33     3 references
34     private void Clear()
35     {
36         CustNameTb.Text = "";
37         CustPhoneTb.Text = "";
38         CustAddTb.Text = "";
39     }
40     1 reference
41     private void SaveBtn_Click(object sender, EventArgs e)
42     {
43         if (CustNameTb.Text == "" || CustAddTb.Text == "" || CustPhoneTb.Text == "")
44         {
45             MessageBox.Show("Missing Information");
46         }
47         else
48         {
49             try
50             {
51                 Con.Open();
52                 SqlCommand cmd = new SqlCommand("insert into CustomersTbl (CustName, CustAdd, CustPhone) values(@CN,@CA,@CP)",
53                 cmd.Parameters.AddWithValue("@CN", CustNameTb.Text);
54                 cmd.Parameters.AddWithValue("@CA", CustAddTb.Text);
55                 cmd.Parameters.AddWithValue("@CP", CustPhoneTb.Text);
56                 cmd.ExecuteNonQuery();
57                 MessageBox.Show("Customer Added");
58                 Con.Close();
59                 DisplayCustomers();
60                 Clear();
61             }
62             catch (Exception Ex)
63             {
64                 MessageBox.Show(Ex.Message);
65             }
66         }
67     }
68 }
```

```
Customers.cs [Design] | Splash.cs [Design]
PetShop | PetShop.Customers | Customers()

65 }
66 int Key = 0;
67 private void CustomersDGV_CellContentClick(object sender, DataGridViewCellEventArgs e)
68 {
69     CustNameTb.Text = CustomersDGV.SelectedRows[0].Cells[1].Value.ToString();
70     CustAddTb.Text = CustomersDGV.SelectedRows[0].Cells[2].Value.ToString();
71     CustPhoneTb.Text = CustomersDGV.SelectedRows[0].Cells[3].Value.ToString();
72     if (CustNameTb.Text == "")
73     {
74         Key = 0;
75     }
76     else
77     {
78         Key = Convert.ToInt32(CustomersDGV.SelectedRows[0].Cells[0].Value.ToString());
79     }
80 }
81
```

```
Customers.cs [Design] | Splash.cs [Design]
PetShop | PetShop.Customers | Customers()

82 private void EditBtn_Click(object sender, EventArgs e)
83 {
84     if (CustNameTb.Text == "" || CustAddTb.Text == "" || CustPhoneTb.Text == "")
85     {
86         MessageBox.Show("Missing Information");
87     }
88     else
89     {
90         try
91         {
92             Con.Open();
93             SqlCommand cmd = new SqlCommand("Update CustomersTbl set CustName=@CN, CustAdd=@CA, CustPhone=@CP where CustID="
94             cmd.Parameters.AddWithValue("@CN", CustNameTb.Text);
95             cmd.Parameters.AddWithValue("@CA", CustAddTb.Text);
96             cmd.Parameters.AddWithValue("@CP", CustPhoneTb.Text);
97             cmd.Parameters.AddWithValue("@CKey", Key);
98             cmd.ExecuteNonQuery();
99             MessageBox.Show("Customer Update");
100             Con.Close();
101             DisplayCustomers();
102             Clear();
103         }
104         catch (Exception Ex)
105         {
106             MessageBox.Show(Ex.Message);
107         }
108     }
109 }
110
```

```
Customers.cs  X Customers.cs [Design]  Splash.cs [Design]
PetShop  PetShop.Customers  Customers()

110
111 1 reference
112 private void DeleteBtn_Click(object sender, EventArgs e)
113 {
114     if (Key == 0)
115     {
116         MessageBox.Show("Select an customer");
117     }
118     else
119     {
120         try
121         {
122             Con.Open();
123             SqlCommand cmd = new SqlCommand("delete from CustomersTbl where CustID = @CKey", Con);
124             cmd.Parameters.AddWithValue("@CKey", Key);
125             cmd.ExecuteNonQuery();
126             MessageBox.Show("Customer Deleted");
127             Con.Close();
128             DisplayCustomers();
129             Clear();
130         }
131         catch (Exception Ex)
132         {
133             MessageBox.Show(Ex.Message);
134         }
135     }
136 }
137
138
139
```


3) Karina Irna Della

a) Uraian Pengerjaan

1. Login Form

- TextBox 'Username' untuk mengisi username yang sesuai.
- TextBox 'Password' untuk mengisi password yang sesuai.
- Button 'Close' di pojok kanan atas digunakan untuk keluar dari halaman Login.
- Button 'LOGIN' digunakan untuk login setelah mengisi username dan password
- Button 'Reset' digunakan untuk menghapus data username dan password yang sudah diisi.

2. Dashboard

- Terdapat progress bar income
- Button 'Dashboard' digunakan untuk menampilkan halaman dashboard.
- Button 'Products' digunakan untuk menampilkan halaman data produk.
- Button 'Employees' digunakan untuk menampilkan halaman data pegawai.
- Button 'Customers' digunakan untuk menampilkan halaman data kustomer/pelanggan.
- Button 'Billing' digunakan untuk menampilkan halaman transaksi.
- Button 'Logout' digunakan untuk keluar dari halaman utama aplikasi dan otomatis kembali ke halaman Login.

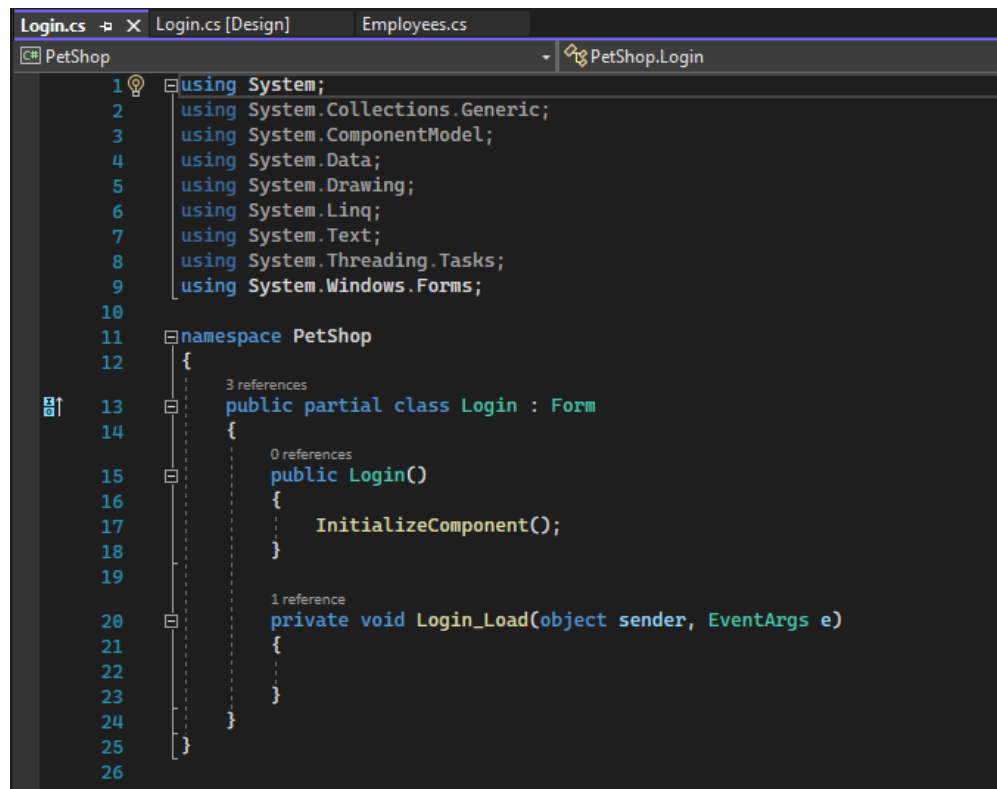
3. Employees Form

- Terdapat beberapa Text box yaitu '**Name**', '**Address**', '**Phone**', dan '**Password**' untuk mengisi data karyawan.
- Date Time Picker untuk mengisi column **Date of Birth** atau Tanggal Lahir karyawan.
- Button 'Save' untuk menyimpan.

- Button 'Edit' untuk mengubah data.
- Button 'Delete' untuk menghapus atau membatalkan penambahan data.
- Data Grid View dengan label 'Employees List' untuk menampilkan data karyawan.

b) Screenshot Kodingan

1. Login Form

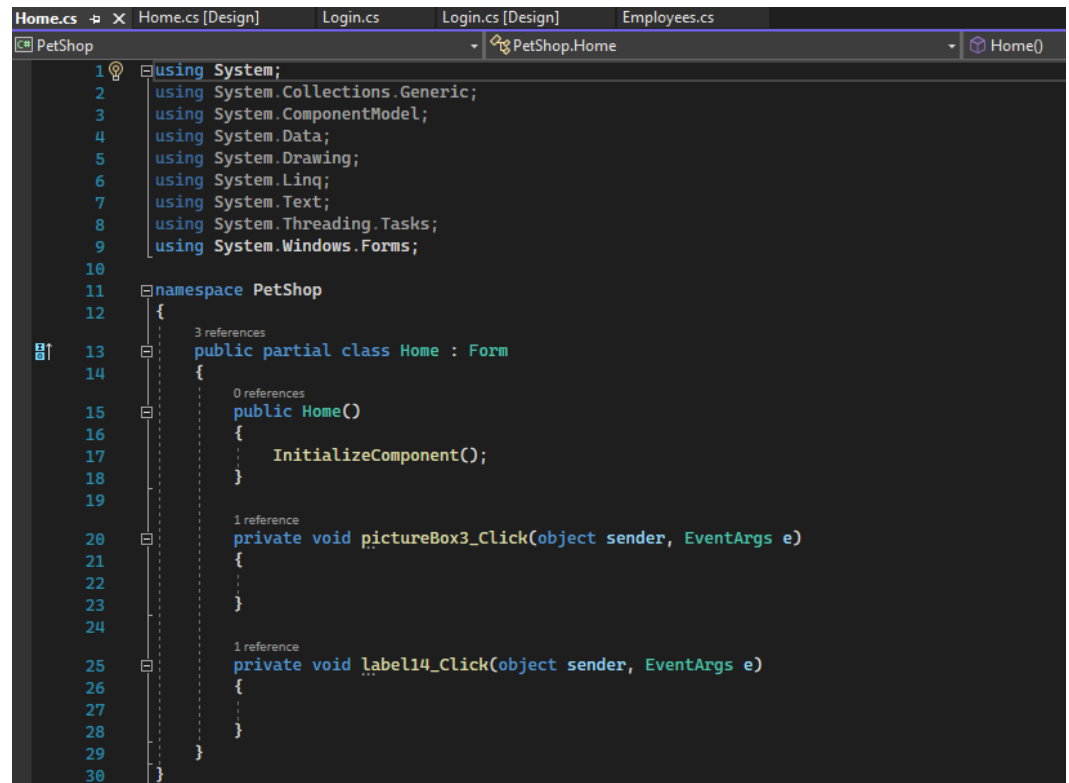


```

Login.cs [Design] Employees.cs
PetShop PetShop.Login

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.Text;
8 using System.Threading.Tasks;
9 using System.Windows.Forms;
10
11 namespace PetShop
12 {
13     public partial class Login : Form
14     {
15         public Login()
16         {
17             InitializeComponent();
18         }
19
20         private void Login_Load(object sender, EventArgs e)
21         {
22         }
23     }
24 }
  
```

2. Dashboard Home

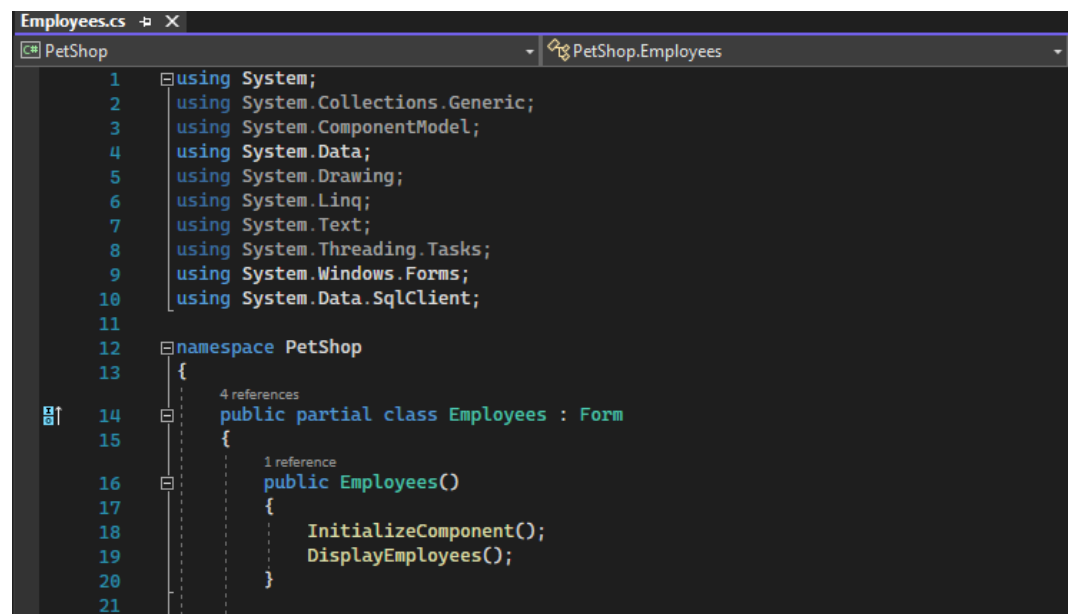


The screenshot shows the Visual Studio IDE with the PetShop project open. The file Home.cs is selected, and the code editor displays the following C# code:

```
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.Text;
8 using System.Threading.Tasks;
9 using System.Windows.Forms;
10
11 namespace PetShop
12 {
13     public partial class Home : Form
14     {
15         public Home()
16         {
17             InitializeComponent();
18         }
19
20         private void pictureBox3_Click(object sender, EventArgs e)
21         {
22         }
23
24         private void label14_Click(object sender, EventArgs e)
25         {
26         }
27     }
28 }
29
30
```

The code defines a partial class `Home` within the `PetShop` namespace, inheriting from `Form`. It includes a constructor `Home()` that calls `InitializeComponent()`. There are two event handler methods: `pictureBox3_Click` and `label14_Click`, both currently empty.

3. Employees Form



The screenshot shows the Visual Studio IDE with the PetShop project open. The file Employees.cs is selected, and the code editor displays the following C# code:

```
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.Text;
8 using System.Threading.Tasks;
9 using System.Windows.Forms;
10 using System.Data.SqlClient;
11
12 namespace PetShop
13 {
14     public partial class Employees : Form
15     {
16         public Employees()
17         {
18             InitializeComponent();
19             DisplayEmployees();
20         }
21     }
22 }
```

The code defines a partial class `Employees` within the `PetShop` namespace, inheriting from `Form`. It includes a constructor `Employees()` that calls `InitializeComponent()` and `DisplayEmployees()`. The `DisplayEmployees()` method is currently empty.

```
Employees.cs PetShop label8_Click(object sender, EventArgs e)
22 private void label8_Click(object sender, EventArgs e)
23 {
24 }
25
26 SqlConnection Con = new SqlConnection(@"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\ASUS\Documents\PetSho
27 private void DisplayEmployees()
28 {
29     Con.Open();
30     string Query = "Select * from EmployeesTbl";
31     SqlDataAdapter sda = new SqlDataAdapter(Query, Con);
32     SqlCommandBuilder Builder = new SqlCommandBuilder(sda);
33     var ds = new DataSet();
34     sda.Fill(ds);
35     EmployeesDGV.DataSource = ds.Tables[0];
36     Con.Close();
37 }
38 private void Clear()
39 {
40     EmpNameTb.Text = "";
41     EmpAddTb.Text = "";
42     EmpPhoneTb.Text = "";
43     PasswordTb.Text = "";
44 }
45
```

```
Employees.cs PetShop label8_Click(object sender, EventArgs e)
45
46 private void SaveBtn_Click(object sender, EventArgs e)
47 {
48     if (EmpNameTb.Text == "" || EmpAddTb.Text == "" || EmpPhoneTb.Text == "" || PasswordTb.Text == "")
49     {
50         MessageBox.Show("Missing Information");
51     }
52     else
53     {
54         try
55         {
56             Con.Open();
57             SqlCommand cmd = new SqlCommand("insert into EmployeesTbl (EmpName, EmpAdd, EmpDOB, EmpPhone, EmpPass) values(
58             cmd.Parameters.AddWithValue("@EN", EmpNameTb.Text);
59             cmd.Parameters.AddWithValue("@EA", EmpAddTb.Text);
60             cmd.Parameters.AddWithValue("@ED", EmpDOB.Value.Date);
61             cmd.Parameters.AddWithValue("@EP", EmpPhoneTb.Text);
62             cmd.Parameters.AddWithValue("@EPa", PasswordTb.Text);
63             cmd.ExecuteNonQuery();
64             MessageBox.Show("Employee Added");
65             Con.Close();
66             DisplayEmployees();
67             Clear();
68         } catch (Exception Ex)
69         {
70             MessageBox.Show(Ex.Message);
71         }
72     }
73 }
74 int Key = 0;
```

```
Employees.cs PetShop PetShop.Employees label8_Click(object sender, EventArgs e)
1 reference
75 private void EmployeesDGV_CellContentClick(object sender, DataGridViewCellEventArgs e)
76 {
77     EmpNameTb.Text = EmployeesDGV.SelectedRows[0].Cells[1].Value.ToString();
78     EmpAddTb.Text = EmployeesDGV.SelectedRows[0].Cells[2].Value.ToString();
79     EmpDOB.Text = EmployeesDGV.SelectedRows[0].Cells[3].Value.ToString();
80     EmpPhoneTb.Text = EmployeesDGV.SelectedRows[0].Cells[4].Value.ToString();
81     PasswordTb.Text = EmployeesDGV.SelectedRows[0].Cells[5].Value.ToString();
82     if (EmpNameTb.Text == "")
83     {
84         Key = 0;
85     }else
86     {
87         Key = Convert.ToInt32(EmployeesDGV.SelectedRows[0].Cells[0].Value.ToString());
88     }
89 }
90
```

```
Employees.cs C:\Users\ASUS\Downloads\PetShop\PetShop\PetShop.Employees.cs PetShop.Employees label8_Click(object sender, EventArgs e)
1 reference
91 private void EditBtn_Click(object sender, EventArgs e)
92 {
93     if (EmpNameTb.Text == "" || EmpAddTb.Text == "" || EmpPhoneTb.Text == "" || PasswordTb.Text == "")
94     {
95         MessageBox.Show("Missing Information");
96     }
97     else
98     {
99         try
100         {
101             Con.Open();
102             SqlCommand cmd = new SqlCommand("Update EmployeesTbl set EmpName=@EN, EmpAdd=@EA, EmpDOB=@ED, EmpPhone=@EP, Em
103             cmd.Parameters.AddWithValue("@EN", EmpNameTb.Text);
104             cmd.Parameters.AddWithValue("@EA", EmpAddTb.Text);
105             cmd.Parameters.AddWithValue("@ED", EmpDOB.Value.Date);
106             cmd.Parameters.AddWithValue("@EP", EmpPhoneTb.Text);
107             cmd.Parameters.AddWithValue("@EPa", PasswordTb.Text);
108             cmd.Parameters.AddWithValue("@EKey", Key);
109             cmd.ExecuteNonQuery();
110             MessageBox.Show("Employee Update");
111             Con.Close();
112             DisplayEmployees();
113             Clear();
114         }
115         catch (Exception Ex)
116         {
117             MessageBox.Show(Ex.Message);
118         }
119     }
120 }
```

```
Employees.cs  X
PetShop  PetShop.Employees  label8_Click(object sender, EventArgs e)

120 }
121
122 1 reference
private void DeleteBtn_Click(object sender, EventArgs e)
123 {
124     if (Key == 0)
125     {
126         MessageBox.Show("Select an employee");
127     }
128     else
129     {
130         try
131         {
132             Con.Open();
133             SqlCommand cmd = new SqlCommand("delete from EmployeesTbl where EmpID = @EKey", Con);
134             cmd.Parameters.AddWithValue("@EKey", Key);
135
136             cmd.ExecuteNonQuery();
137             MessageBox.Show("Employee Deleted");
138             Con.Close();
139             DisplayEmployees();
140             Clear();
141         }
142         catch (Exception Ex)
143         {
144             MessageBox.Show(Ex.Message);
145         }
146     }
147 }
148
```

```
Employees.cs  X
PetShop  US\Downloads\PetShop\PetShop\PetShop\Employees  PetShop.Employees  label8_Click(object sender, EventArgs e)

149 1 reference
private void label4_Click(object sender, EventArgs e)
150 {
151     Customers Obj = new Customers();
152     Obj.Show();
153     this.Hide();
154 }
155
156 1 reference
private void label1_Click(object sender, EventArgs e)
157 {
158     Products Obj = new Products();
159     Obj.Show();
160     this.Hide();
161 }
162
163 1 reference
private void label3_Click(object sender, EventArgs e)
164 {
165 }
166 }
167 }
168 }
169
```