

BAB VI Kesimpulan dan Saran

VI.1 Simpulan

Proyek Tugas Akhir dalam bentuk pengembangan program / aplikasi ini dapat ditarik kesimpulan yaitu:

- Secara garis besar, aplikasi pembukuan telah dapat memenuhi kebutuhan untuk melakukan pencatatan, perubahan, penghapusan dan pencarian data.
- Aplikasi pembukuan ini telah dapat membuat berbagai laporan dasar seperti laporan buku besar, laporan neraca saldo, laporan pembukuan, laporan konsumen, laporan produsen, laporan karyawan, laporan barang, laporan bahan baku, laporan inventaris, laporan produksi bahan, laporan produksi barang, laporan pengambilan bahan, laporan pengambilan barang, laporan penggajian, laporan kas bon, dan laporan history.
- Aplikasi pembukuan ini juga cukup mudah digunakan karena user interface yang mudah dipahami.
- Aplikasi pembukuan ini juga terdapat *error handling* yang dapat memberitahu karyawan apabila terdapat kesalahan dalam memasukkan data.
- Aplikasi pembukuan terdapat fitur *log in* dengan menggunakan password yang membuat aplikasi aman digunakan dengan mencegah orang yang tidak berwenang untuk masuk ke dalam aplikasi ini.

VI.2 Saran

Penulis ingin memberikan saran dalam pelaksanaan Tugas Akhir dalam bentuk pengembangan aplikasi yaitu :

1. Perlu adanya pelatihan untuk karyawan (khususnya pegawai yang mengani pembukuan) agar dapat bekerja secara optimal.
2. Sistem Aplikasi Pembukuan CV. Taja Mitra Lestari sebaiknya melakukan pendataan data barang yang dimilikinya secara bertahap sehingga tidak mengganggu proses penjualan yang sedang berlangsung saat ini.

DAFTAR PUSTAKA

- [Deitel 2002] Deitel. (2002). **C# How To Program**, Prentice Hall.
- [Horngren 2006] Horngren, dkk. (2006). **Akuntansi**, jilid 1. edisi 6. Indeks, Jakarta. ISBN 979-683-757-9.
- [Jason 2003] Jason Price. (2003). **Mastering C# Database Programming**, Sybex.
- [Kevin 2002] Kevin Kline. (2002). **SQL in a Nutshell**, O'Reilly.
- [Meigs 1970] Meigs, Walter B. and Robert F. Meigs. (1970). **Financial Accounting, 4th ed**, McGraw-Hill, p.1. ISBN 0-07-041534-X (old edition).
- [Mcgraw] Mcgraw Hill Osborn. **Programming in C#. Net**.
- [Stig 2004] Stig Sæther Bakken. (2004). **PHP Manual**.
- [Sugiarto 2002] Sugiarto. (2002). **Pengantar Akuntansi, Pusat Penerbitan Universitas Terbuka**, Jakarta. ISBN 979-689-057-7.

LAMPIRAN

Source Code

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace AplikasiPembukuan
{
    public partial class MainForm : Form
    {
        public static string karyawanId;
        public static string karyawanGrup;

        private int childFormNumber = 0;

        public MainForm()
        {
            InitializeComponent();
        }

        private void TutupMenu()
        {
            loginToolStripMenuItem.Enabled = true;
            logoutToolStripMenuItem.Enabled = false;
            pembukuanToolStripMenuItem.Enabled = false;
            kodeAkunToolStripMenuItem.Enabled = false;
            lihatTransaksiPembelianToolStripMenuItem.Enabled =
false;

            tambahTransaksiPembelianBahanBakuToolStripMenuItem.Enabled =
false;

            tambahTransaksiPembelianBarangToolStripMenuItem.Enabled = false;

            tambahTransaksiPembelianInventarisToolStripMenuItem.Enabled =
false;

            lihatTransaksiPenjualanToolStripMenuItem.Enabled =
false;

            tambahTransaksiPenjualanToolStripMenuItem.Enabled =
false;

            lihatPengambilanBahanBakuToolStripMenuItem.Enabled =
false;

            lihatPengambilanBarangToolStripMenuItem.Enabled =
false;

            tambahPengambilanBahanBakuToolStripMenuItem.Enabled =
false;

            tambahPengambilanBarangToolStripMenuItem.Enabled =
false;
        }
    }
}
```

```

        lihatProduksiBahanBakuToolStripMenuItem.Enabled =
false;
        lihatProduksiBarangToolStripMenuItem.Enabled = false;
        tambahProduksiBahanBakuToolStripMenuItem.Enabled =
false;
        tambahProduksiBarangToolStripMenuItem.Enabled =
false;
        bahanBakuToolStripMenuItem.Enabled = false;
        barangToolStripMenuItem.Enabled = false;
        inventarisToolStripMenuItem.Enabled = false;
        konsumenToolStripMenuItem.Enabled = false;
        produsenToolStripMenuItem.Enabled = false;
        karyawanToolStripMenuItem.Enabled = false;
        lihatPenggajianToolStripMenuItem.Enabled = false;
        lihatKasBonToolStripMenuItem.Enabled = false;
        tambahPenggajianToolStripMenuItem.Enabled = false;
        tambahKasBonToolStripMenuItem.Enabled = false;
        historyToolStripMenuItem.Enabled = false;
        ubahPasswordToolStripMenuItem.Enabled = false;
        bukuBesarToolStripMenuItem.Enabled = false;
        laporanPembukuanToolStripMenuItem.Enabled = false;
        laporanPembelianToolStripMenuItem.Enabled = false;
        laporanPembelianBahanBakuToolStripMenuItem.Enabled =
false;
        laporanPembelianBarangToolStripMenuItem.Enabled =
false;
        laporanPembelianInventarisToolStripMenuItem.Enabled =
false;
        laporanPenjualanToolStripMenuItem.Enabled = false;
        laporanPenjualanBarangToolStripMenuItem.Enabled =
false;
        laporanPengambilanBahanBakuToolStripMenuItem.Enabled
= false;
        laporanPengambilanBarangToolStripMenuItem.Enabled =
false;
        laporanProduksiBahanBakuToolStripMenuItem.Enabled =
false;
        laporanProduksiBarangToolStripMenuItem.Enabled =
false;
        laporanBahanBakuToolStripMenuItem.Enabled = false;
        laporanBarangToolStripMenuItem.Enabled = false;
        laporanInventarisToolStripMenuItem.Enabled = false;
        laporanKonsumenToolStripMenuItem.Enabled = false;
        laporanProdusenToolStripMenuItem.Enabled = false;
        laporanKaryawanToolStripMenuItem.Enabled = false;
        laporanPenggajianToolStripMenuItem.Enabled = false;
        laporanKasBonToolStripMenuItem.Enabled = false;
        laporanHistoryToolStripMenuItem.Enabled = false;
    }

    LoginForm logFrm = new LoginForm();

    private void loginToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        logFrm.ShowDialog(this);

        if (logFrm.Logged == true)
        {

```

```

        karyawanId = logFrm.Id;
        karyawanGrup = logFrm.Grup;
        logoutToolStripMenuItem.Enabled = true;
        loginToolStripMenuItem.Enabled = false;
        ubahPasswordToolStripMenuItem.Enabled = true;
        StatusLbl.Text = "Selamat Datang " + logFrm.Id;

    this.modul_grupTbTableAdapter.SearchModul(this.pembukuanDataSet.m
    odul_grupTb, logFrm.Id);

        int jumlah =
    (int)pembukuanDataSet.Tables["modul_grupTb"].Rows.Count;

        for (int i = 0; i < jumlah; i++)
        {
            string idModul =
    pembukuanDataSet.Tables["modul_grupTb"].Rows[i]["id_modul"].ToStr
    ing();

            switch (idModul)
            {
                case "main pembukuan":
                    pembukuanToolStripMenuItem.Enabled =
    true;
                    break;
                case "main kode akun":
                    kodeAkunToolStripMenuItem.Enabled =
    true;
                    break;
                case "main transaksi pembelian":
                    lihatTransaksiPembelianToolStripMenuItem.Enabled = true;
                    break;
                case "tambah pembelian bahan baku":
                    tambahTransaksiPembelianBahanBakuToolStripMenuItem.Enabled =
    true;
                    break;
                case "tambah pembelian barang":
                    tambahTransaksiPembelianBarangToolStripMenuItem.Enabled = true;
                    break;
                case "tambah pembelian inventaris":
                    tambahTransaksiPembelianInventarisToolStripMenuItem.Enabled =
    true;
                    break;
                case "main transaksi penjualan":
                    lihatTransaksiPenjualanToolStripMenuItem.Enabled = true;
                    break;
                case "tambah penjualan barang":
                    tambahTransaksiPenjualanToolStripMenuItem.Enabled = true;
                    break;
                case "main pengambilan bahan baku":
                    lihatPengambilanBahanBakuToolStripMenuItem.Enabled = true;

```

```

        break;
        case "main pengambilan barang":
lihatPengambilanBarangToolStripMenuItem.Enabled = true;
        break;
        case "tambah pengambilan bahan baku":
tambahPengambilanBahanBakuToolStripMenuItem.Enabled = true;
        break;
        case "tambah pengambilan barang":
tambahPengambilanBarangToolStripMenuItem.Enabled = true;
        break;
        case "main produksi bahan baku":
lihatProduksiBahanBakuToolStripMenuItem.Enabled = true;
        break;
        case "main produksi barang":
lihatProduksiBarangToolStripMenuItem.Enabled = true;
        break;
        case "tambah produksi bahan baku":
tambahProduksiBahanBakuToolStripMenuItem.Enabled = true;
        break;
        case "tambah produksi barang":
tambahProduksiBarangToolStripMenuItem.Enabled = true;
        break;
        case "main bahan baku":
        bahanBakuToolStripMenuItem.Enabled =
true;
        break;
        case "main barang":
        barangToolStripMenuItem.Enabled =
true;
        break;
        case "main inventaris":
        inventarisToolStripMenuItem.Enabled =
true;
        break;
        case "main konsumen":
        konsumenToolStripMenuItem.Enabled =
true;
        break;
        case "main produsen":
        produsenToolStripMenuItem.Enabled =
true;
        break;
        case "main karyawan":
        karyawanToolStripMenuItem.Enabled =
true;
        break;
        case "main penggajian":
lihatPenggajianToolStripMenuItem.Enabled = true;
        break;
        case "main kas bon":

```

```

        lihatKasBonToolStripMenuItem.Enabled
= true;
        break;
        case "tambah penggajian":
tambahPenggajianToolStripMenuItem.Enabled = true;
        break;
        case "tambah kas bon":
        tambahKasBonToolStripMenuItem.Enabled
= true;
        break;
        case "main history":
        historyToolStripMenuItem.Enabled =
true;
        break;
        case "laporan neraca":
        bukuBesarToolStripMenuItem.Enabled =
true;
        break;
        case "laporan pembukuan":
laporanPembukuanToolStripMenuItem.Enabled = true;
        break;
        case "laporan pembelian":
laporanPembelianToolStripMenuItem.Enabled = true;
        break;
        case "laporan pembelian bahan baku":
laporanPembelianBahanBakuToolStripMenuItem.Enabled = true;
        break;
        case "laporan pembelian barang":
laporanPembelianBarangToolStripMenuItem.Enabled = true;
        break;
        case "laporan pembelian inventaris":
laporanPembelianInventarisToolStripMenuItem.Enabled = true;
        break;
        case "laporan penjualan":
laporanPenjualanToolStripMenuItem.Enabled = true;
        break;
        case "laporan penjualan barang":
laporanPenjualanBarangToolStripMenuItem.Enabled = true;
        break;
        case "laporan pengambilan bahan baku":
laporanPengambilanBahanBakuToolStripMenuItem.Enabled = true;
        break;
        case "laporan pengambilan barang":
laporanPengambilanBarangToolStripMenuItem.Enabled = true;
        break;
        case "laporan produksi bahan baku":
laporanProduksiBahanBakuToolStripMenuItem.Enabled = true;
        break;

```



```

        case "laporan produksi barang":
laporanProduksiBarangToolStripMenuItem.Enabled = true;
        break;
        case "laporan bahan baku":
laporanBahanBakuToolStripMenuItem.Enabled = true;
        break;
        case "laporan barang":
laporanBarangToolStripMenuItem.Enabled = true;
        break;
        case "laporan inventaris":
laporanInventarisToolStripMenuItem.Enabled = true;
        break;
        case "laporan konsumen":
laporanKonsumenToolStripMenuItem.Enabled = true;
        break;
        case "laporan produsen":
laporanProdusenToolStripMenuItem.Enabled = true;
        break;
        case "laporan karyawan":
laporanKaryawanToolStripMenuItem.Enabled = true;
        break;
        case "laporan penggajian":
laporanPenggajianToolStripMenuItem.Enabled = true;
        break;
        case "laporan kas bon":
laporanKasBonToolStripMenuItem.Enabled = true;
        break;
        case "laporan history":
laporanHistoryToolStripMenuItem.Enabled = true;
        break;
        default:
        break;
    }
}
}
}

private void CascadeToolStripMenuItem_Click(object
sender, EventArgs e)
{
    LayoutMdi(MdiLayout.Cascade);
}

private void TileVerticalToolStripMenuItem_Click(object
sender, EventArgs e)
{
    LayoutMdi(MdiLayout.TileVertical);
}

```

```

private void TileHorizontalToolStripMenuItem_Click(object
sender, EventArgs e)
{
    LayoutMdi(MdiLayout.TileHorizontal);
}

private void CloseAllToolStripMenuItem_Click(object
sender, EventArgs e)
{
    foreach (Form childForm in MdiChildren)
    {
        childForm.Close();
    }
}

private void logoutToolStripMenuItem_Click(object sender,
EventArgs e)
{
    foreach (Form childForm in MdiChildren)
    {
        childForm.Close();
    }
    TutupMenu();
    StatusLbl.Text = "Selamat Datang Pada Aplikasi
Pembukuan..";
    logFrm.Id = "";
    logFrm.Password = "";
}

private void MainForm_Load(object sender, EventArgs e)
{
    // TODO: This line of code loads data into the
'pembukuanDataSet.modul_grupTb' table. You can move, or remove
it, as needed.

this.modul_grupTbTableAdapter.Fill(this.pembukuanDataSet.modul_gr
upTb);

    // TODO: This line of code loads data into the
'pembukuanDataSet.modul_grupTb' table. You can move, or remove
it, as needed.

this.modul_grupTbTableAdapter.Fill(this.pembukuanDataSet.modul_gr
upTb);

    //TutupMenu();
    StatusLbl.Text = "Selamat Datang Pada Aplikasi
Pembukuan..";
}

private void keluarToolStripMenuItem_Click(object sender,
EventArgs e)
{
    foreach (Form childForm in MdiChildren)
    {
        childForm.Close();
    }
    this.Close();
}

```

```

private void produsenToolStripMenuItem_Click(object
sender, EventArgs e)
{
    MainProdusenForm.Show(this);
}

private void konsumenToolStripMenuItem_Click(object
sender, EventArgs e)
{
    MainKonsumenForm.Show(this);
}

private void karyawanToolStripMenuItem_Click(object
sender, EventArgs e)
{
    MainKaryawanForm.Show(this);
}

private void
aboutPembukuanV12ToolStripMenuItem_Click(object sender, EventArgs
e)
{
    AboutBox.Show(this);
}

private void
tambahTransaksiPembelianBahanBakuToolStripMenuItem_Click(object
sender, EventArgs e)
{
    TambahPembelianBahanBakuForm.Show(this);
}

private void bahanBakuToolStripMenuItem_Click(object
sender, EventArgs e)
{
    MainBahanBakuForm.Show(this);
}

private void
tambahTransaksiPembelianBarangToolStripMenuItem_Click(object
sender, EventArgs e)
{
    TambahPembelianBarangForm.Show(this);
}

private void barangToolStripMenuItem_Click(object sender,
EventArgs e)
{
    MainBarangForm.Show(this);
}

private void
tambahTransaksiPembelianInventarisToolStripMenuItem_Click(object
sender, EventArgs e)
{
    TambahPembelianInventarisForm.Show(this);
}

```

```

        private void inventarisToolStripMenuItem_Click(object
sender, EventArgs e)
        {
            MainInventarisForm.Show(this);
        }

        private void
tambahProduksiBahanBakuToolStripMenuItem_Click(object sender,
EventArgs e)
        {
            TambahProduksiBahanForm.Show(this);
        }

        private void
lihatProduksiBahanToolStripMenuItem_Click(object sender,
EventArgs e)
        {
            MainProduksiBahanForm.Show(this);
        }

        private void
tambahProduksiBarangToolStripMenuItem_Click(object sender,
EventArgs e)
        {
            TambahProduksiBarangForm.Show(this);
        }

        private void ubahPasswordToolStripMenuItem_Click(object
sender, EventArgs e)
        {
            UbahPasswordForm.Show(this);
        }

        private void historyToolStripMenuItem_Click(object
sender, EventArgs e)
        {
            MainHistoryForm.Show(this);
        }

        private void
lihatProduksiBarangToolStripMenuItem_Click(object sender,
EventArgs e)
        {
            MainProduksiBarangForm.Show(this);
        }

        private void
lihatPenggajianToolStripMenuItem_Click(object sender, EventArgs
e)
        {
            MainPenggajianForm.Show(this);
        }

        private void
tambahPenggajianToolStripMenuItem_Click(object sender, EventArgs
e)
        {
            TambahPenggajianForm.Show(this);
        }

```

```

        private void
tambahTransaksiPenjualanToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        TambahPenjualanBarangForm.Show(this);
    }

        private void
tambahPengambilanBahanToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        TambahPengambilanBahanForm.Show(this);
    }

        private void
tambahPengambilanBarangToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        TambahPengambilanBarangForm.Show(this);
    }

        private void
lihatTransaksiPembelianToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        MainPembelianForm.Show(this);
    }

        private void
lihatTransaksiPenjualanToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        MainPenjualanForm.Show(this);
    }

        private void
lihatPengambilanBahanToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        MainPengambilanBahanForm.Show(this);
    }

        private void
lihatPengambilanBarangToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        MainPengambilanBarangForm.Show(this);
    }

        private void pembukuanToolStripMenuItem_Click(object
sender, EventArgs e)
    {
        MainPembukuanForm.Show(this);
    }

        private void kodeAkunToolStripMenuItem_Click(object
sender, EventArgs e)
    {

```

```

        MainKodeAkunForm.Show(this);
    }

    private void tambahKasBonToolStripMenuItem_Click(object
sender, EventArgs e)
    {
        TambahKasBonForm.Show(this);
    }

    private void lihatKasBonToolStripMenuItem_Click(object
sender, EventArgs e)
    {
        MainKasBonForm.Show(this);
    }

    private void
laporanKonsumenToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        LaporanKonsumenForm.Show(this);
    }

    private void
laporanPembukuanToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        LaporanPembukuanForm.Show(this);
    }

    private void
laporanPembelianToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        LaporanPembelianForm.Show(this);
    }

    private void
laporanPenjualanToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        LaporanPenjualanForm.Show(this);
    }

    private void
laporanBahanBakuToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        LaporanBahanBakuForm.Show(this);
    }

    private void laporanBarangToolStripMenuItem_Click(object
sender, EventArgs e)
    {
        LaporanBarangForm.Show(this);
    }

```

```

        private void
laporanInventarisToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        LaporanInventarisForm.Show(this);
    }

        private void
laporanProdusenToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        LaporanProdusenForm.Show(this);
    }

        private void
laporanPembelianBahanBakuToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        LaporanPembelianBahanForm.Show(this);
    }

        private void
laporanPenjualanBarangToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        LaporanPenjualanBarangForm.Show(this);
    }

        private void
laporanKaryawanToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        LaporanKaryawanForm.Show(this);
    }

        private void
laporanPengambilanBahanBakuToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        LaporanPengambilanBahanForm.Show(this);
    }

        private void laporanHistoryToolStripMenuItem_Click(object
sender, EventArgs e)
    {
        LaporanHistoryForm.Show(this);
    }

        private void
laporanPembelianBarangToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        LaporanPembelianBarangForm.Show(this);
    }

        private void
laporanPembelianInventarisToolStripMenuItem_Click(object sender,
EventArgs e)
    {

```

```

        LaporanPembelianInventarisForm.Show(this);
    }

    private void
laporanPengambilanBarangToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        LaporanPengambilanBarangForm.Show(this);
    }

    private void
laporanProduksiBahanBakuToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        LaporanProduksiBahanForm.Show(this);
    }

    private void
laporanProduksiBarangToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        LaporanProduksiBarangForm.Show(this);
    }

    private void
laporanPenggajianToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        LaporanPenggajianForm.Show(this);
    }

    private void laporanKasBonToolStripMenuItem_Click(object
sender, EventArgs e)
    {
        LaporanKasBonForm.Show(this);
    }

    private void bukuBesarKasToolStripMenuItem_Click(object
sender, EventArgs e)
    {
        LaporanBukuBesarKasForm.Show(this);
    }

    private void neracaSaldoToolStripMenuItem_Click(object
sender, EventArgs e)
    {
        LaporanNeracaSaldoForm.Show(this);
    }

    private void neracaToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        LaporanNeracaForm.Show(this);
    }

    private void
bukuBesarPiutangToolStripMenuItem_Click(object sender, EventArgs
e)
    {

```



```

        LaporanBukuBesarPiutangForm.Show(this);
    }

    private void
bukuBesarHutangToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        LaporanBukuBesarHutangForm.Show(this);
    }

    private void
bukuBesarPenjualanToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        LaporanBukuBesarPenjualanForm.Show(this);
    }

    private void
bukuBesarPembelianToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        LaporanBukuBesarPembelianForm.Show(this);
    }

    private void
bukuBesarBebanGajiToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        LaporanBukuBesarBebanGajiForm.Show(this);
    }
}

```

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace AplikasiPembukuan
{
    public partial class MainKaryawanForm : Form
    {
        public MainKaryawanForm()
        {
            InitializeComponent();
        }

        private static MainKaryawanForm karyaFrm;

        public static void Show(Form mdiParent)
        {
            if (karyaFrm == null)
                karyaFrm = new MainKaryawanForm();
        }
    }
}

```

```

        karyaFrm.MdiParent = mdiParent;
        karyaFrm.Show();
    }

    protected override void OnClosed(EventArgs e)
    {
        base.OnClosed(e);
        MainKaryawanForm.karyaFrm = null;
    }

    private string idKaryawan;

    public static bool appId;

    public string IdKaryawan
    {
        get { return idKaryawan; }
        set { idKaryawan = value; }
    }

    private int gajiPokok;

    public int GajiPokok
    {
        get { return gajiPokok; }
        set { gajiPokok = value; }
    }

    private void MainKaryawanForm_Load(object sender,
EventArgs e)
    {
        // TODO: This line of code loads data into the
        'pembukuanDataSet.historyTb' table. You can move, or remove it,
        as needed.

        this.historyTbTableAdapter.Fill(this.pembukuanDataSet.historyTb);
        // TODO: This line of code loads data into the
        'pembukuanDataSet.pembukuanTb' table. You can move, or remove it,
        as needed.

        this.pembukuanTbTableAdapter.Fill(this.pembukuanDataSet.pembukuan
Tb);
        // TODO: This line of code loads data into the
        'pembukuanDataSet.grupTb' table. You can move, or remove it, as
        needed.

        this.grupTbTableAdapter.Fill(this.pembukuanDataSet.grupTb);
        // TODO: This line of code loads data into the
        'pembukuanDataSet.karyawanTb' table. You can move, or remove it,
        as needed.

        this.karyawanTbTableAdapter.Fill(this.pembukuanDataSet.karyawanTb
);

        if (MainForm.karyawanGrup == "Administrator" ||
MainForm.karyawanGrup == "Employee Manager")
        {
            hapusToolStripButton.Enabled = true;
            ubahToolStripButton.Enabled = true;
        }
    }

```

```

    }
    else if (MainForm.karyawanGrup == null)
    {
        hapusToolStripButton.Enabled = false;
        ubahToolStripButton.Enabled = false;
    }
    else
    {
        hapusToolStripButton.Enabled = false;
        ubahToolStripButton.Enabled = false;
    }

    if (appId == true)
    {
        okBtn.Visible = true;
        okBtn.Enabled = true;
    }
}

public string max_id()
{
    int max = 0;
    string max_fin = "";

    int jumlah =
pembukuanDataSet.Tables["karyawanTb"].Rows.Count;

    for (int i = 0; i < jumlah; i++)
    {
        string id =
pembukuanDataSet.Tables["karyawanTb"].Rows[i]["id"].ToString();
        int lenght = id.Length;
        string y = id.Substring(3, lenght - 3);
        int z;
        int.TryParse(y, out z);

        if (max < z)
        {
            max = z;
        }
    }

    max = max + 1;

    if (max < 10)
    {
        max_fin = "00000" + max.ToString();
    }
    else if (max < 100)
    {
        max_fin = "0000" + max.ToString();
    }
    else if (max < 1000)
    {
        max_fin = "000" + max.ToString();
    }
    else if (max < 10000)
    {
        max_fin = "00" + max.ToString();
    }
}

```

```

    }
    else if (max < 100000)
    {
        max_fin = "0" + max.ToString();
    }
    else if (max < 1000000)
    {
        max_fin = "" + max.ToString();
    }
    else
    {
        errorProvider.SetError(tambahIdTxt, "nomor sudah
penuh. hubungi administrator anda");
    }

    return max_fin;
}

public void kosongkanTambah()
{
    tambahNamaTxt.Text = "";
    tambahAlamatTxt.Text = "";
    tambahTeleponTxt.Text = "";
    tambahNoKtpTxt.Text = "";
    tambahPasswordTxt.Text = "";
    tambahGajiPokokTxt.Text = "";
}

private void hapusToolStripButton_Click(object sender,
EventArgs e)
{
    if (karyawanTbBindingSource.Current != null)
    {
        DialogResult buttonPressed =
MessageBox.Show("Apakah anda yakin akan menghapus karyawan ini?",
"Konfirmasi", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);
        if (buttonPressed == DialogResult.Yes)
        {
            try
            {
                karyawanTbBindingSource.RemoveCurrent();

karyawanTbTableAdapter.Update(pembukuanDataSet.karyawanTb);
            }
            catch
            {
                errorProvider.SetError(cariLbl, "Id
karyawan ini masih dipakai.");
            }
        }
    }
}

private void cariBtn_Click(object sender, EventArgs e)
{
    if (kategoriCB.SelectedIndex == 8)
    {
        if (minTanggalDTP.Value > maxTanggalDTP.Value)

```

```

        errorProvider.SetError(cariLbl, "minimal
lebih besar dari maksimal");
    else
    {
        try
        {
            karyawanTbBindingSource.Filter =
("tanggal_lahir >= '" + minTanggalDTP.Value + "' and
tanggal_lahir <= '" + maxTanggalDTP.Value + "'");
            errorProvider.Clear();
        }
        catch
        {
            errorProvider.SetError(cariLbl, "harus
diisi dengan angka");
        }
    }
}
else if (kategoriCB.SelectedIndex == 9)
{
    int minT, maxT;
    int.TryParse(minTxt.Text, out minT);
    int.TryParse(maxTxt.Text, out maxT);
    if (minT > maxT)
        errorProvider.SetError(cariLbl, "minimal
lebih besar dari maksimal");
    else
    {
        try
        {
            karyawanTbBindingSource.Filter =
("gaji_pokok >= '" + minTxt.Text + "' and gaji_pokok <= '" +
maxTxt.Text + "'");
            errorProvider.Clear();
        }
        catch
        {
            errorProvider.SetError(cariLbl, "harus
diisi dengan angka");
            minTxt.Text = "";
            maxTxt.Text = "";
        }
    }
}
else if (kategoriCB.SelectedIndex == 10)
{
    int minT, maxT;
    int.TryParse(minTxt.Text, out minT);
    int.TryParse(maxTxt.Text, out maxT);
    if (minT > maxT)
        errorProvider.SetError(cariLbl, "minimal
lebih besar dari maksimal");
    else
    {
        try
        {
            karyawanTbBindingSource.Filter = ("utang
>= '" + minTxt.Text + "' and utang <= '" + maxTxt.Text + "'");
            errorProvider.Clear();
        }
    }
}

```

```

        }
        catch
        {
            errorProvider.SetError(cariLbl, "harus
diisi dengan angka");
            minTxt.Text = "";
            maxTxt.Text = "";
        }
    }
}

private void tambahBtn_Click(object sender, EventArgs e)
{
    if (tambahNamaTxt.Text != "" || tambahAlamatTxt.Text
!= "" || tambahTeleponTxt.Text != "")
    {
        try
        {
            this.karyawanTbTableAdapter.Insert(tambahIdTxt.Text,
tambahPasswordTxt.Text, tambahGrupCB.SelectedValue.ToString(),
tambahNamaTxt.Text,
tambahAlamatTxt.Text, tambahKotaCB.SelectedItem.ToString(),
tambahTeleponTxt.Text, tambahNoKtpTxt.Text, tambahTanggalDTP.Value
, int.Parse(tambahGajiPokokTxt.Text),
int.Parse(tambahUtangTxt.Text));

            this.karyawanTbTableAdapter.Fill(pembukuanDataSet.karyawanTb);

            string max = max_id();

            tambahIdTxt.Text = "KR-" + max;

            errorProvider.Clear();
            kosongkanTambah();
        }
        catch
        {
            errorProvider.SetError(tambahIdTxt, "error");
        }
        else
        {
            errorProvider.SetError(tambahIdTxt, "textbox yang
bertanda * harus diisi.");
        }
    }

    private void tambahToolStripButton_Click(object sender,
EventArgs e)
    {
        if (tambahPanel.Visible == true)
        {
            tambahPanel.Visible = false;
            this.Height = 310;
        }
        else
        {

```

```

        tambahPanel.Visible = true;
        this.Height = 512;

        kosongkanTambah();

        string max = max_id();

        tambahIdTxt.Text = "KR-" + max;

        errorProvider.Clear();
    }
}

private void tambahKosongkanBtn_Click(object sender,
EventArgs e)
{
    kosongkanTambah();
}

private void cariTxt_TextChanged(object sender, EventArgs
e)
{
    if (cariTxt.Text == string.Empty)
        karyawanTbBindingSource.RemoveFilter();
    else if (kategoriCB.SelectedIndex == 0)
    {
        try
        {
            karyawanTbBindingSource.Filter = ("id like
'%" + cariTxt.Text + "%'");
            errorProvider.Clear();
        }
        catch
        {
            cariTxt.Text = "";
        }
    }
    else if (kategoriCB.SelectedIndex == 1)
    {
        try
        {
            karyawanTbBindingSource.Filter = ("password
like '%" + cariTxt.Text + "%'");
            errorProvider.Clear();
        }
        catch
        {
            cariTxt.Text = "";
        }
    }
    else if (kategoriCB.SelectedIndex == 3)
    {
        try
        {
            karyawanTbBindingSource.Filter = ("nama like
'%" + cariTxt.Text + "%'");
            errorProvider.Clear();
        }
        catch

```

```

        {
            cariTxt.Text = "";
        }
    }
    else if (kategoriCB.SelectedIndex == 4)
    {
        try
        {
            karyawanTbBindingSource.Filter = ("alamat
like '%" + cariTxt.Text + "%'");
            errorProvider.Clear();
        }
        catch
        {
            cariTxt.Text = "";
        }
    }
    else if (kategoriCB.SelectedIndex == 5)
    {
        try
        {
            karyawanTbBindingSource.Filter = ("kota like
 '%" + cariTxt.Text + "%'");
            errorProvider.Clear();
        }
        catch
        {
            cariTxt.Text = "";
        }
    }
    else if (kategoriCB.SelectedIndex == 6)
    {
        try
        {
            karyawanTbBindingSource.Filter = ("telepon
like '%" + cariTxt.Text + "%'");
            errorProvider.Clear();
        }
        catch
        {
            cariTxt.Text = "";
        }
    }
    else if (kategoriCB.SelectedIndex == 7)
    {
        try
        {
            karyawanTbBindingSource.Filter = ("no_ktp
like '%" + cariTxt.Text + "%'");
            errorProvider.Clear();
        }
        catch
        {
            cariTxt.Text = "";
        }
    }
}
}

```



```

private void kategoriCB_SelectedIndexChanged(object
sender, EventArgs e)
{
    if (kategoriCB.SelectedIndex == 0)
    {
        karyawanTbBindingSource.RemoveFilter();
        minTxt.Visible = false;
        maxTxt.Visible = false;
        labelsd.Visible = false;
        cariBtn.Visible = false;
        cariTxt.Visible = true;
        minTanggalDTP.Visible = false;
        maxTanggalDTP.Visible = false;
        cariTxt.Text = "";
        minTxt.Text = "";
        maxTxt.Text = "";
    }
    else if (kategoriCB.SelectedIndex == 1)
    {
        karyawanTbBindingSource.RemoveFilter();
        minTxt.Visible = false;
        maxTxt.Visible = false;
        labelsd.Visible = false;
        cariBtn.Visible = false;
        cariTxt.Visible = true;
        minTanggalDTP.Visible = false;
        maxTanggalDTP.Visible = false;
        cariTxt.Text = "";
        minTxt.Text = "";
        maxTxt.Text = "";
    }
    else if (kategoriCB.SelectedIndex == 2)
    {
        karyawanTbBindingSource.RemoveFilter();
        minTxt.Visible = false;
        maxTxt.Visible = false;
        labelsd.Visible = false;
        cariBtn.Visible = false;
        cariTxt.Visible = false;
        cariGrupCB.Visible = true;
        minTanggalDTP.Visible = false;
        maxTanggalDTP.Visible = false;
        cariTxt.Text = "";
        minTxt.Text = "";
        maxTxt.Text = "";
    }
    else if (kategoriCB.SelectedIndex == 3)
    {
        karyawanTbBindingSource.RemoveFilter();
        minTxt.Visible = false;
        maxTxt.Visible = false;
        labelsd.Visible = false;
        cariBtn.Visible = false;
        cariTxt.Visible = true;
        cariGrupCB.Visible = false;
        minTanggalDTP.Visible = false;
        maxTanggalDTP.Visible = false;
        cariTxt.Text = "";
        minTxt.Text = "";
    }
}

```

```

        maxTxt.Text = "";
    }
    else if (kategoriCB.SelectedIndex == 4)
    {
        karyawanTbBindingSource.RemoveFilter();
        minTxt.Visible = false;
        maxTxt.Visible = false;
        labelsd.Visible = false;
        cariBtn.Visible = false;
        cariTxt.Visible = true;
        cariGrupCB.Visible = false;
        minTanggalDTP.Visible = false;
        maxTanggalDTP.Visible = false;
        cariTxt.Text = "";
        minTxt.Text = "";
        maxTxt.Text = "";
    }
    else if (kategoriCB.SelectedIndex == 5)
    {
        karyawanTbBindingSource.RemoveFilter();
        minTxt.Visible = false;
        maxTxt.Visible = false;
        labelsd.Visible = false;
        cariBtn.Visible = false;
        cariTxt.Visible = true;
        cariGrupCB.Visible = false;
        minTanggalDTP.Visible = false;
        maxTanggalDTP.Visible = false;
        cariTxt.Text = "";
        minTxt.Text = "";
        maxTxt.Text = "";
    }
    else if (kategoriCB.SelectedIndex == 6)
    {
        karyawanTbBindingSource.RemoveFilter();
        minTxt.Visible = false;
        maxTxt.Visible = false;
        labelsd.Visible = false;
        cariBtn.Visible = false;
        cariTxt.Visible = true;
        cariGrupCB.Visible = false;
        minTanggalDTP.Visible = false;
        maxTanggalDTP.Visible = false;
        cariTxt.Text = "";
        minTxt.Text = "";
        maxTxt.Text = "";
    }
    else if (kategoriCB.SelectedIndex == 7)
    {
        karyawanTbBindingSource.RemoveFilter();
        minTxt.Visible = false;
        maxTxt.Visible = false;
        labelsd.Visible = false;
        cariBtn.Visible = false;
        cariTxt.Visible = true;
        cariGrupCB.Visible = false;
        minTanggalDTP.Visible = false;
        maxTanggalDTP.Visible = false;
        cariTxt.Text = "";
    }

```

```

        minTxt.Text = "";
        maxTxt.Text = "";
    }
    else if (kategoriCB.SelectedIndex == 8)
    {
        karyawanTbBindingSource.RemoveFilter();
        minTxt.Visible = false;
        maxTxt.Visible = false;
        labelsd.Visible = true;
        cariBtn.Visible = true;
        cariTxt.Visible = false;
        cariGrupCB.Visible = false;
        minTanggalDTP.Visible = true;
        maxTanggalDTP.Visible = true;
        cariTxt.Text = "";
        minTxt.Text = "";
        maxTxt.Text = "";
    }
    else if (kategoriCB.SelectedIndex == 9)
    {
        karyawanTbBindingSource.RemoveFilter();
        minTxt.Visible = true;
        maxTxt.Visible = true;
        labelsd.Visible = true;
        cariBtn.Visible = true;
        cariTxt.Visible = false;
        cariGrupCB.Visible = false;
        minTanggalDTP.Visible = false;
        maxTanggalDTP.Visible = false;
        cariTxt.Text = "";
        minTxt.Text = "";
        maxTxt.Text = "";
    }
    else if (kategoriCB.SelectedIndex == 10)
    {
        karyawanTbBindingSource.RemoveFilter();
        minTxt.Visible = true;
        maxTxt.Visible = true;
        labelsd.Visible = true;
        cariBtn.Visible = true;
        cariTxt.Visible = false;
        cariGrupCB.Visible = false;
        minTanggalDTP.Visible = false;
        maxTanggalDTP.Visible = false;
        cariTxt.Text = "";
        minTxt.Text = "";
        maxTxt.Text = "";
    }
}

private void ubahToolStripButton_Click(object sender,
EventArgs e)
{
    UbahKaryawanForm ubahKaryaFrm = new
    UbahKaryawanForm();

    DataRowView rowView = karyawanTbBindingSource.Current
as DataRowView;

```

```

        pembukuanDataSet.karyawanTbRow row = rowView.Row as
pembukuanDataSet.karyawanTbRow;

        ubahKaryaFrm.Id = row.id;
        ubahKaryaFrm.Password = row.password;
        ubahKaryaFrm.Grup = row.id_grup;
        ubahKaryaFrm>Nama = row.nama;
        ubahKaryaFrm.Alatnat = row.alamat;
        ubahKaryaFrm.Kota = row.kota;
        ubahKaryaFrm.Telp = row.telepon;
        ubahKaryaFrm.No_ktp = row.no_ktp;
        ubahKaryaFrm.Tanggal = row.tanggal_lahir;
        ubahKaryaFrm.Gaji_Pokok = row.gaji_pokok;
        ubahKaryaFrm.Utang = row.utang;

        ubahKaryaFrm.ShowDialog();

        if (ubahKaryaFrm.CancelApp != false)
        {
            row.id=ubahKaryaFrm.Id;
            row.password=ubahKaryaFrm.Password;
            row.id_grup=ubahKaryaFrm.Grup;
            row.nama=ubahKaryaFrm>Nama;
            row.alamat=ubahKaryaFrm.Alatnat;
            row.kota = ubahKaryaFrm.Kota;
            row.telepon=ubahKaryaFrm.Telp;
            row.no_ktp=ubahKaryaFrm.No_ktp;
            row.tanggal_lahir=ubahKaryaFrm.Tanggal;
            row.gaji_pokok=ubahKaryaFrm.Gaji_Pokok;

            this.karyawanTbTableAdapter.Update(pembukuanDataSet.karyawanTb);

            this.karyawanTbTableAdapter.Fill(pembukuanDataSet.karyawanTb);
        }

        private void cariGrupCB_SelectedIndexChanged(object
sender, EventArgs e)
        {
            if (kategoriCB.SelectedIndex == 2)
            {
                try
                {
                    karyawanTbBindingSource.Filter = ("id_grup
like '%" + cariGrupCB.SelectedValue.ToString() + "%'");
                    errorProvider.Clear();
                }
                catch
                {
                    cariTxt.Text = "";
                }
            }
        }

        private void okBtn_Click(object sender, EventArgs e)
        {
            if (karyawanTbBindingSource.Current != null)
            {

```

```

        DataRowView rowView =
karyawanTbBindingSource.Current as DataRowView;
        pembukuanDataSet.karyawanTbRow row = rowView.Row
as pembukuanDataSet.karyawanTbRow;

        idKaryawan = row.id;
        gajiPokok = row.gaji_pokok;
        appId = false;
    }
}

private void pembayaranKasBonToolStripButton_Click(object
sender, EventArgs e)
{
    PembayaranKasBonForm pembayaranKasBonFrm = new
PembayaranKasBonForm();

    DataRowView rowView = karyawanTbBindingSource.Current
as DataRowView;
    pembukuanDataSet.karyawanTbRow row = rowView.Row as
pembukuanDataSet.karyawanTbRow;

    pembayaranKasBonFrm.IdKaryawan = row.id;
    pembayaranKasBonFrm.Utang = row.utang;

    pembayaranKasBonFrm.ShowDialog();

    if (pembayaranKasBonFrm.CancelApp != false)
    {
        string keterangan;
        int pelunasan;

        row.utang = pembayaranKasBonFrm.Utang;
        pelunasan = pembayaranKasBonFrm.Pelunasan;

        keterangan = "Pembayaran hutang karyawan " +
row.id + " sebanyak " + pelunasan.ToString();

this.karyawanTbTableAdapter.Update(pembukuanDataSet.karyawanTb);

this.karyawanTbTableAdapter.Fill(pembukuanDataSet.karyawanTb);

        this.pembukuanTbTableAdapter.Insert("101",
MainForm.karyawanId, "", System.DateTime.Now, keterangan,
pelunasan, 0);
        this.pembukuanTbTableAdapter.Insert("102",
MainForm.karyawanId, "", System.DateTime.Now, keterangan, 0,
pelunasan);

this.pembukuanTbTableAdapter.Fill(pembukuanDataSet.pembukuanTb);

this.historyTbTableAdapter.Insert(MainForm.karyawanId,
System.DateTime.Now, keterangan);

this.historyTbTableAdapter.Fill(pembukuanDataSet.historyTb);
    }
}

```

```

    }
}

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace AplikasiPembukuan
{
    public partial class MainPengambilanBahanForm : Form
    {
        public MainPengambilanBahanForm()
        {
            InitializeComponent();
        }

        private void MainPengambilanBahanForm_Load(object sender,
EventArgs e)
        {
            // TODO: This line of code loads data into the
'pembukuanDataSet.dpengambilan_bahanTb' table. You can move, or
remove it, as needed.

this.dpengambilan_bahanTbTableAdapter.Fill(this.pembukuanDataSet.
dpengambilan_bahanTb);
            // TODO: This line of code loads data into the
'pembukuanDataSet.mpengambilan_bahanTb' table. You can move, or
remove it, as needed.

this.mpengambilan_bahanTbTableAdapter.Fill(this.pembukuanDataSet.
mpengambilan_bahanTb);

        }

        private static MainPengambilanBahanForm
mainPengambilanBahanFrm;

        public static void Show(Form mdiParent)
        {
            if (mainPengambilanBahanFrm == null)
                mainPengambilanBahanFrm = new
MainPengambilanBahanForm();

            mainPengambilanBahanFrm.MdiParent = mdiParent;
            mainPengambilanBahanFrm.Show();
        }

        protected override void OnClosed(EventArgs e)
        {
            base.OnClosed(e);
            MainPengambilanBahanForm.mainPengambilanBahanFrm =
null;
        }
    }
}

```

```

private void cariBtn_Click(object sender, EventArgs e)
{
    if (kategoriCB.SelectedIndex == 2)
    {
        if (minTanggalDTP.Value > maxTanggalDTP.Value)
            errorProvider.SetError(cariLbl, "minimal
lebih besar dari maksimal");
        else
        {
            try
            {
                mpengambilanbahanTbBindingSource.Filter =
("tanggal >= " + minTanggalDTP.Value + " and tanggal <= " +
maxTanggalDTP.Value + "");
                errorProvider.Clear();
            }
            catch
            {
                errorProvider.SetError(cariLbl, "harus
diisi dengan angka");
            }
        }
    }
}

private void cariTxt_TextChanged(object sender, EventArgs
e)
{
    if (cariTxt.Text == string.Empty)
        mpengambilanbahanTbBindingSource.RemoveFilter();
    else if (kategoriCB.SelectedIndex == 0)
    {
        try
        {
            mpengambilanbahanTbBindingSource.Filter =
("id like '%" + cariTxt.Text + "%'");
            errorProvider.Clear();
        }
        catch
        {
            cariTxt.Text = "";
        }
    }
    else if (kategoriCB.SelectedIndex == 1)
    {
        try
        {
            mpengambilanbahanTbBindingSource.Filter =
("id_karyawan like '%" + cariTxt.Text + "%'");
            errorProvider.Clear();
        }
        catch
        {
            cariTxt.Text = "";
        }
    }
    else if (kategoriCB.SelectedIndex == 3)
    {

```

```

        try
        {
            mpengambilanbahanTbBindingSource.Filter =
("keterangan like '%" + cariTxt.Text + "%'");
            errorProvider.Clear();
        }
        catch
        {
            cariTxt.Text = "";
        }
    }

private void kategoriCB_SelectedIndexChanged(object
sender, EventArgs e)
{
    if (kategoriCB.SelectedIndex == 0)
    {
        mpengambilanbahanTbBindingSource.RemoveFilter();
        minTanggalDTP.Visible = false;
        maxTanggalDTP.Visible = false;
        labelsd.Visible = false;
        cariBtn.Visible = false;
        cariTxt.Visible = true;
        cariTxt.Text = "";
    }
    else if (kategoriCB.SelectedIndex == 1)
    {
        mpengambilanbahanTbBindingSource.RemoveFilter();
        minTanggalDTP.Visible = false;
        maxTanggalDTP.Visible = false;
        labelsd.Visible = false;
        cariBtn.Visible = false;
        cariTxt.Visible = true;
        cariTxt.Text = "";
    }
    else if (kategoriCB.SelectedIndex == 2)
    {
        mpengambilanbahanTbBindingSource.RemoveFilter();
        minTanggalDTP.Visible = true;
        maxTanggalDTP.Visible = true;
        labelsd.Visible = true;
        cariBtn.Visible = true;
        cariTxt.Visible = false;
        cariTxt.Text = "";
    }
    else if (kategoriCB.SelectedIndex == 3)
    {
        mpengambilanbahanTbBindingSource.RemoveFilter();
        minTanggalDTP.Visible = false;
        maxTanggalDTP.Visible = false;
        labelsd.Visible = false;
        cariBtn.Visible = false;
        cariTxt.Visible = true;
        cariTxt.Text = "";
    }
}
}

```



```

}

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace AplikasiPembukuan
{
    public partial class LaporanBukuBesarPembelianForm : Form
    {
        public LaporanBukuBesarPembelianForm()
        {
            InitializeComponent();
        }

        private void cariBtn_Click(object sender, EventArgs e)
        {
            if (minTanggalDTP.Value > maxTanggalDTP.Value)
                errorProvider.SetError(cariLbl, "minimal lebih
besar dari maksimal");
            else
            {
                try
                {

                    this.pembukuanTbTableAdapter.SearchByDate(pembukuanDataSet.pembuk
uanTb, minTanggalDTP.Value, maxTanggalDTP.Value);

                    BukuBesarPembelianCrystalReport.SetDataSource(pembukuanDataSet);
                    crystalReportViewer.RefreshReport();
                    errorProvider.Clear();
                }
                catch
                {
                    errorProvider.SetError(cariLbl, "error");
                }
            }
        }

        private static LaporanBukuBesarPembelianForm
laporanBukuBesarPembelianFrm;

        public static void Show(Form mdiParent)
        {
            if (laporanBukuBesarPembelianFrm == null)
                laporanBukuBesarPembelianFrm = new
LaporanBukuBesarPembelianForm();

            laporanBukuBesarPembelianFrm.MdiParent = mdiParent;
            laporanBukuBesarPembelianFrm.Show();
        }

        protected override void OnClosed(EventArgs e)

```

```

        {
            base.OnClosed(e);
        }

        LaporanBukuBesarPembelianForm.laporanBukuBesarPembelianFrm =
        null;
    }

    private void LaporanBukuBesarPembelianForm_Load(object
sender, EventArgs e)
    {
        // TODO: This line of code loads data into the
        'pembukuanDataSet.pembukuanTb' table. You can move, or remove it,
        as needed.

        this.pembukuanTbTableAdapter.Fill(this.pembukuanDataSet.pembukuan
Tb);

        BukuBesarPembelianCrystalReport.SetDataSource(pembukuanDataSet);
        crystalReportViewer.RefreshReport();
    }
}

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace AplikasiPembukuan
{
    public partial class TambahPengambilanBahanForm : Form
    {
        public TambahPengambilanBahanForm()
        {
            InitializeComponent();
        }

        private void TambahPengambilanBahanForm_Load(object
sender, EventArgs e)
        {
            // TODO: This line of code loads data into the
            'pembukuanDataSet.dpengambilan_bahanTb' table. You can move, or
            remove it, as needed.

            this.dpengambilan_bahanTbTableAdapter.Fill(this.pembukuanDataSet.
dpengambilan_bahanTb);

            // TODO: This line of code loads data into the
            'pembukuanDataSet.bahan_bakuTb' table. You can move, or remove
            it, as needed.

            this.bahan_bakuTbTableAdapter.Fill(this.pembukuanDataSet.bahan_ba
kuTb);
        }
    }
}

```

```

        // TODO: This line of code loads data into the
        'pembukuanDataSet.mpengambilan_bahanTb' table. You can move, or
        remove it, as needed.

        this.mpengambilan_bahanTbTableAdapter.Fill(this.pembukuanDataSet.
        mpengambilan_bahanTb);

        string max = max_id();

        idKaryawanTxt.Text = MainForm.karyawanId;
        idMasterTxt.Text = "PK-" + max;

        mpengambilan_bahanTbTableAdapter.Insert(idMasterTxt.Text, null,
        null, null);

        mpengambilan_bahanTbTableAdapter.Fill(pembukuanDataSet.mpengambil
        an_bahanTb);

        dpengambilanbahanTbBindingSource.Filter = ("id_master
        = '" + idMasterTxt.Text + "'");

        tanggalLbl.Text =
        System.DateTime.Now.ToShortDateString();
    }

    public int totalAwal, totalAkhir;
    public decimal jumlahGudang;

    private static TambahPengambilanBahanForm
    tambahPengambilanBahanFrm;

    public static void Show(Form mdiParent)
    {
        if (tambahPengambilanBahanFrm == null)
            tambahPengambilanBahanFrm = new
            TambahPengambilanBahanForm();

        tambahPengambilanBahanFrm.MdiParent = mdiParent;
        tambahPengambilanBahanFrm.Show();
    }

    protected override void OnClosed(EventArgs e)
    {
        base.OnClosed(e);
        TambahPengambilanBahanForm.tambahPengambilanBahanFrm
        = null;
    }

    public int max_id_detail()
    {
        int max = 0;

        int jumlah =
        pembukuanDataSet.Tables["dpengambilan_bahanTb"].Rows.Count;

        for (int i = 0; i < jumlah; i++)
        {

```

```

        int id =
int.Parse(pembukuanDataSet.Tables["dpengambilan_bahanTb"].Rows[i]
["id"].ToString());

        if (max < id)
        {
            max = id;
        }
    }

    max = max + 1;

    return max;
}

public string max_id()
{
    int max = 0;
    string max_fin = "";

    int jumlah =
pembukuanDataSet.Tables["mpengambilan_bahanTb"].Rows.Count;

    for (int i = 0; i < jumlah; i++)
    {
        string id =
pembukuanDataSet.Tables["mpengambilan_bahanTb"].Rows[i]["id"].ToS
tring();

        int lenght = id.Length;
        string y = id.Substring(3, lenght - 3);
        int z;
        int.TryParse(y, out z);

        if (max < z)
        {
            max = z;
        }
    }

    max = max + 1;

    if (max < 10)
    {
        max_fin = "00000" + max.ToString();
    }
    else if (max < 100)
    {
        max_fin = "0000" + max.ToString();
    }
    else if (max < 1000)
    {
        max_fin = "000" + max.ToString();
    }
    else if (max < 10000)
    {
        max_fin = "00" + max.ToString();
    }
    else if (max < 100000)
    {

```

```

        max_fin = "0" + max.ToString();
    }
    else if (max < 1000000)
    {
        max_fin = "" + max.ToString();
    }
    else
    {
        errorProvider.SetError(idMasterTxt, "nomor sudah
penuh. hubungi administrator anda");
    }

    return max_fin;
}

private void tambahBtn_Click(object sender, EventArgs e)
{
    decimal jumlah;
    decimal.TryParse(jumlahTxt.Text, out jumlah);
    if (jumlahGudang < jumlah)
    {
        errorProvider.Clear();
        errorProvider.SetError(idMasterTxt, "jumlah tidak
mencukupi");
    }
    else
    {
        try
        {
            dpengambilan_bahanTbTableAdapter.Insert(idMasterTxt.Text,
max_id_detail(), idBahanBakuTxt.Text,
decimal.Parse(jumlahTxt.Text));

            dpengambilan_bahanTbTableAdapter.Fill(pembukuanDataSet.dpengambil
an_bahanTb);

            idBahanBakuTxt.Text = "";
            jumlahTxt.Text = "";
        }
        catch
        {
            errorProvider.SetError(idMasterTxt,
"Error.");
        }
    }
}

private void idBahanBakuBtn_Click(object sender,
EventArgs e)
{
    MainBahanBakuForm mainBahanBakuFrm = new
MainBahanBakuForm();
    MainBahanBakuForm.appId = true;
    mainBahanBakuFrm.ShowDialog();

    if (mainBahanBakuFrm.DialogResult == DialogResult.OK)
    {

```

```

        idBahanBakuTxt.Text =
mainBahanBakuFrm.IdBahanBaku;
        jumlahGudang = mainBahanBakuFrm.JumlahGudang;
    }
}

private void hapusBtn_Click(object sender, EventArgs e)
{
    if (dpengambilanbahanTbBindingSource.Current != null)
    {
        DataRowView rowView =
dpengambilanbahanTbBindingSource.Current as DataRowView;
        pembukuanDataSet.dpengambilan_bahanTbRow row =
rowView.Row as pembukuanDataSet.dpengambilan_bahanTbRow;

        DialogResult buttonPressed =
MessageBox.Show("Apakah anda yakin akan menghapus pengambilan
ini?", "Konfirmasi", MessageBoxButtons.YesNo,
MessageBoxIcon.Warning);
        if (buttonPressed == DialogResult.Yes)
        {
            try
            {

dpengambilanbahanTbBindingSource.RemoveCurrent();

dpengambilan_bahanTbTableAdapter.Update(pembukuanDataSet.dpengamb
ilan_bahanTb);
                }
                catch
                {
                    errorProvider.SetError(idMasterTxt,
"Error.");
                }
            }
        }
    }

private void okBtn_Click(object sender, EventArgs e)
{
    try
    {
        errorProvider.Clear();

mpengambilan_bahanTbTableAdapter.Update(idMasterTxt.Text,
idKaryawanTxt.Text, System.DateTime.Now,
keteranganTxt.Text, idMasterTxt.Text, null, null, null);

mpengambilan_bahanTbTableAdapter.Fill(pembukuanDataSet.mpengambil
an_bahanTb);

        this.Close();
    }
    catch
    {
        errorProvider.Clear();
        errorProvider.SetError(idMasterTxt, "error");
    }
}

```

```

        private void batalBtn_Click(object sender, EventArgs e)
        {
            try
            {
                errorProvider.Clear();

                this.dpengambilan_bahanTbTableAdapter.SearchId(this.pembukuanDataSet.dpengambilan_bahanTb, idMasterTxt.Text);
                int jumlah =
                pembukuanDataSet.Tables["dpengambilan_bahanTb"].Rows.Count;

                for (int i = 0; i <= jumlah - 1; i++)
                {
                    string id =
                    pembukuanDataSet.Tables["dpengambilan_bahanTb"].Rows[i]["id"].ToString();

                    dpengambilan_bahanTbTableAdapter.DeleteDetail(int.Parse(id));
                }

                dpengambilan_bahanTbTableAdapter.Fill(pembukuanDataSet.dpengambilan_bahanTb);

                mpengambilan_bahanTbTableAdapter.DeleteMaster(idMasterTxt.Text);

                mpengambilan_bahanTbTableAdapter.Fill(pembukuanDataSet.mpengambilan_bahanTb);

                this.Close();
            }
            catch
            {
                errorProvider.SetError(idMasterTxt, "error");
            }
        }
    }

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace AplikasiPembukuan
{
    public partial class UbahPembelianForm : Form
    {
        public UbahPembelianForm()
    }
}

```

```

{
    InitializeComponent();
}

public int sisaPembayaran;
public int pelunasan;

private int totalPembayaran;

public int TotalPembayaran
{
    get { return totalPembayaran; }
    set { totalPembayaran = value; }
}

public string Id
{
    get { return idTxt.Text; }
    set { idTxt.Text = value; }
}

public string IdKaryawan
{
    get { return idKaryawanTxt.Text; }
    set { idKaryawanTxt.Text = value; }
}

public string IdProdusen
{
    get { return idProdusenTxt.Text; }
    set { idProdusenTxt.Text = value; }
}

public string Tanggal
{
    get { return tanggalTxt.Text; }
    set { tanggalTxt.Text = value; }
}

private int total;

public int Total
{
    get { return int.Parse(totalTxt.Text); }
    set
    {
        totalTxt.Text = value.ToString();
        total = value;
    }
}

private int pembayaran;

public int Pembayaran
{
    get { return int.Parse(pembayaranTxt.Text); }
    set
    {
        pembayaranTxt.Text = value.ToString();
    }
}

```



```

        pembayaran = value;
    }
}
private string statusLunas;

public string StatusLunas
{
    get { return statusLunas; }
    set { statusLunas = value; }
}
private string statusKirim;

public string StatusKirim
{
    get { return statusKirim; }
    set { statusKirim = value; }
}

public string Bagian
{
    get { return bagianLbl.Text; }
    set { bagianLbl.Text = value; }
}

public bool cancelApp;

public bool CancelApp
{
    get { return cancelApp; }
    set { cancelApp = value; }
}

private void ubahBtn_Click(object sender, EventArgs e)
{
    totalPembayaran = pembayaran + pelunasan;
    if (totalPembayaran > total)
    {
        cancelApp = false;
        errorProvider.SetError(idTxt, "pelunasan lebih
besar dari sisa pembayaran");
    }
    else
    {
        try
        {
            cancelApp = true;
            this.Close();
        }
        catch
        {
            errorProvider.SetError(idTxt, "error");
        }
    }
}

private void batalBtn_Click(object sender, EventArgs e)
{
    cancelApp = false;
}

```

```
        this.Close();
    }

    private void UbahPembelianForm_Load(object sender,
EventArgs e)
    {
        sisaPembayaran = total - pembayaran;
        sisaPembayaranTxt.Text = sisaPembayaran.ToString();

        if (statusKirim == "belum terkirim")
            kirimCheckBox.Checked = false;
        else if (statusKirim == "sudah terkirim")
            kirimCheckBox.Checked = true;

        if (statusLunas == "belum lunas")
            lunasCheckBox.Checked = false;
        else if (statusLunas == "sudah lunas")
            lunasCheckBox.Checked = true;
    }

    private void pelunasanTxt_TextChanged(object sender,
EventArgs e)
    {
        int.TryParse(pelunasanTxt.Text, out pelunasan);
        sisaPembayaran = total - pembayaran - pelunasan;
        sisaPembayaranTxt.Text = sisaPembayaran.ToString();

        if (sisaPembayaran == 0)
        {
            statusLunas = "sudah lunas";
            lunasCheckBox.Checked = true;
        }
        else
        {
            statusLunas = "belum lunas";
            lunasCheckBox.Checked = false;
        }
    }

    private void kirimCheckBox_CheckedChanged(object sender,
EventArgs e)
    {
        if (kirimCheckBox.Checked == true)
            statusKirim = "sudah terkirim";
        else if (kirimCheckBox.Checked == false)
            statusKirim = "belum terkirim";
    }
}
}
```