

SİSTEM ANALİZİ VE TASARIM DERSİ

Sinema Otomasyonu

DERSİN ÖĞRETİM ELEMANI

DOÇ. DR. AŞİR ÖZBEK

Hakan Arslan

200705022 NÖ

KIRIKKALE – 2021

İçindekiler

[Şekiller 3](#_Toc100328244)

[1.Akış Şemaları 4](#_Toc100329072)

[1.1 Giriş Ekranı Akış Şeması 4](#_Toc100329073)

[1.2 Salon Ekle Akış Şeması 5](#_Toc100329074)

[1.3 Film Ekle Akış Şeması 6](#_Toc100329075)

[1.4 Seans Ekle Akış Şeması 7](#_Toc100329076)

[1.5 Seans Listele Akış Şeması 8](#_Toc100329077)

[1.6 Satış Listele Akış Şeması 9](#_Toc100329078)

[1.7 Bilet Satış Akış Şeması 10](#_Toc100329079)

[1.8 Bilet İptal Akış Şeması 11](#_Toc100329080)

[2. C# Form Tasarımı Ve Kodları 12](#_Toc100329081)

[2.1 Ana Sayfa / Gösterge Paneli Tasarımı 15](#_Toc100329082)

[2.2 Salon Ekle Sayfa Tasarımı 22](#_Toc100329083)

[2.3 Film Ekle Sayfa Tasarımı 24](#_Toc100329084)

[2.4 Seans Ekle Sayfa Tasarımı 27](#_Toc100329085)

[2.5 Seans Listele Sayfa Tasarımı 31](#_Toc100329086)

[2.6 Satış Listele Sayfa Tasarımı 34](#_Toc100329087)

[2.7 Bilet Satış Sayfa Tasarımı 37](#_Toc100329088)

Şekiller

**1)**[Giriş Ekranı Akış Şeması 3](#_Toc100328244)

[**2)** Salon Ekle Akış Şeması 4](#_Toc100329072)

[**3)** Film Ekle Akış Şeması ı 5](#_Toc100329074)

[**4)** Seans Ekle Akış Şeması 6](#_Toc100329075)

[**5)** Seans Listele Akış Şeması 7](#_Toc100329076)

[**6)** Satış Listele Akış Şeması 8](#_Toc100329077)

[**7)** Bilet Satış Akış Şeması 9](#_Toc100329078)

[**8)** Bilet İptal Akış Şeması 10](#_Toc100329079)

[**9)** Yetkili Girişi 11](#_Toc100329080)

[**10)** Yetkili Girişi - Başarısız Giriş 13](#_Toc100329081)

[**11)** Yetkili Girişi – Başarılı Giriş 13](#_Toc100329082)

[**12)** Anasayfa Tasarımı 14](#_Toc100329083)

[**13)** Anasayfa Tasarımı 2 19](#_Toc100329084)

[**14)** Salon Ekle Tasarımı 21](#_Toc100329085)

[**15)** Salon Ekle – Hata Mesajı 22](#_Toc100329086)

[**16)** Film Ekle Tasarımı 23](#_Toc100329087)

[**17)** Film Ekle – Afiş Seçimi 23](#_Toc100329088)

**18)** Film Ekle – Veri Girişi**…………………………………………………………………………..**24

**19)** Film Ekle – Onay Mesajı**……………………………………………………………………..**24

**20)** Film Ekle –Hata Mesajı **……………………………………………………………………..**24

**21)** Seans Ekle Tasarımı **…………………………………………………………………………..**26

**22)** Seans Ekle – Onay Mesajı **……………………………………………………………………..**27

**23)** Seans Ekle – Hata Mesajı **……………………………………………………………………..**27

**24)** Seans Liste Tasarımı **…………………………………………………………………………..**30

**25)** Seans Liste Tasarımı 2 **………………………………………………………………………..**31

**26)** Seans Liste Tasarımı 3 **………………………………………………………………………..**31

**27)** Satış Listele Tasarımı **………………………………………………………………………..**33

**28)** Satış Listele Tasarımı 2**………………………………………………………………………..**34

**29)** Satış Listele Tasarımı 3 **………………………………………………………………………..**34

**30)** Bilet Satış Tasarımı **………………………………………………………………………..**36

**31)** Bilet Satış Tasarımı 2 **………………………………………………………………………..37**

**32)** Bilet Satış Tasarımı 3**………………………………………………………………………..38**

# 1.Akış Şemaları

# Giriş Ekranı Akış Şeması

**Yetkili Giriş Ekranı akış diyagramı**

**Bilgileri**

**Kontrol et**

**GİRİŞ YAP BUTONUNA TIKLA**

**ŞİFRE GİR**

**KULLANICI ADI GİR**

Hayır

**Uyarı  
Kullanıcı Adı Veya Şifre Hatalı**

Evet

**Anasayfa formunu göster**

**Şekil 1:** Giriş Ekranı Akış Şeması

# Salon Ekle Akış Şeması

**Salon Ekleye Tıkla**

**Salon Adı Gir**

**Verileri Veri Tabanına Aktar**

**Mesaj  
Salon Eklendi!**

**Salon İsimlerini Kontrol Et**

Evet

Hayır

**Uyarı  
Bu salon zaten mevcut!**

**Şekil 2:** Salon Ekle Akış Şeması

# Film Ekle Akış Şeması

**Film Ekle Tıkla**

**Film Bilgilerini Gir**

**Verileri Veri Tabanına Aktar**

**Mesaj  
Başarıyla Kayıt Yapıldı**

**Film Bilgilerini Kontrol et**

Hayır

Evet

**Uyarı  
Bu film zaten mevcut!**

**Şekil 3:** Film Ekle Akış Şeması

# Seans Ekle Akış Şeması

**Seans Ekle Tıkla**

**Film salon ve seans saati seç**

**Verileri Veri Tabanına Aktar**

**Mesaj  
Başarıyla Kayıt Yapıldı**

**Seçilen Tarihi Kontrol Et**

Hayır

Evet

**Uyarı  
Geçmişe Dönük İşlem yapılamaz**

**Şekil 4:** Seans Ekle Akış Şeması

# Seans Listele Akış Şeması

**Form Yüklenirken Veri Tabanından Kayıtları Çek**

**Kayıtları Data Grid Viewe Ata**

**Şekil 5:** Seans Listele Akış Şeması

# Satış Listele Akış Şeması

**Form Yüklenirken Veri Tabanından Kayıtları Çek**

**Kayıtları Data Grid Viewe Ata**

**Şekil 6:** Satış Listele Akış Şeması

# Bilet Satış Akış Şeması

**Film adı, Salon adı, Film tarihi, Film Seansı, Koltuk No, Ad, Soyad, Ücret seç**

**Seçilen/Girilen bilgileri veri tabanında ilgili alanlara kaydet**

**Mesaj  
Bilet Kesildi!**

**Bilet Sat Tıkla**

**Şekil 7:** Bilet Satış Akış Şeması

# Bilet İptal Akış Şeması

**Satın alınmış koltuk bilgilerini veri tabanından getir**

**Seçilen Koltuk Numarasını veri tabanından sil**

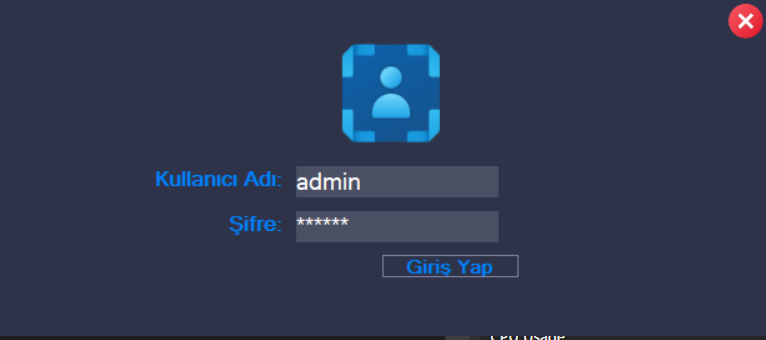
**Mesaj  
Bilet Kesildi!**

**Koltuk No Seç Tıkla**

**Şekil 8:** Bilet İptal Akış Şeması

# 2. C# Form Tasarımı Ve Kodları

Tasarıma geçmeden önce örnek projeleri inceledik ve yapacağımız tasarımın günümüz projelerine ayak uydurabilmesi adına tasarımda modernleşmeye gittik. İlk olarak sisteme dışarıdan müdahaleyi engellemek adına bir yönetici giriş sayfası oluşturduk. Giriş bilgileri için kullanıcı adı ve şifre belirlenmiştir. Eğer girilen bilgiler veri tabanındaki yönetici bilgileri ile eşleşirse sisteme giriş yapılacak şekilde kodlanmıştır.



**Şekil 9:** Yetkili Girişi

**Form Bilgileri:**

Form İsmi : login

Kullanıcı adı girişi : textbox name textBox1

Şifre girişi : textbox name textBox2

Giriş yap : buton name button1

Form bu şekilde ayarlanmıştır. Giriş butonu ile ilgili kodlarımız aşağıdaki gibi yazılmıştır. Bilgilerin doğruluğuna ve yanlışlığına göre uyarı kutucukları eklenmiştir.

public partial class login : Form

{

SqlConnection bag = new SqlConnection("Data Source=DESKTOP-RTSGMFH;Initial Catalog=SinemaOtomasyonu;Integrated Security=True");

public login()

{

InitializeComponent();

}

private void login\_Load(object sender, EventArgs e)

{

}

private void pictureBox1\_Click(object sender, EventArgs e)

{

var reply = MessageBox.Show("Çıkış yapmak istiyor musunuz?", "Çıkış talebinde bulundunuz", MessageBoxButtons.YesNo);

DialogResult sonuc = reply;

if (sonuc == DialogResult.Yes)

{

Application.Exit();

}

}

bool isThere;

private void button1\_Click(object sender, EventArgs e)

{

string k\_adi = textBox1.Text;

string sifre = textBox2.Text;

bag.Open();

SqlCommand command = new SqlCommand("select \* from admins",bag);

SqlDataReader reader = command.ExecuteReader();

while (reader.Read())

{

if (k\_adi==reader["k\_adi"].ToString().TrimEnd() && sifre==reader["sifre"].ToString().TrimEnd())

{

isThere = true;

break;

}

else

{

isThere = false;

}

}

bag.Close();

if (isThere)

{

MessageBox.Show("Tekrardan hoşgeldin"+textBox1.Text+"!","Giriş Yapıldı");

this.Hide();

anasayfa test = new anasayfa();

test.usernamefromlogin = textBox1.Text;

test.Show();

}

else

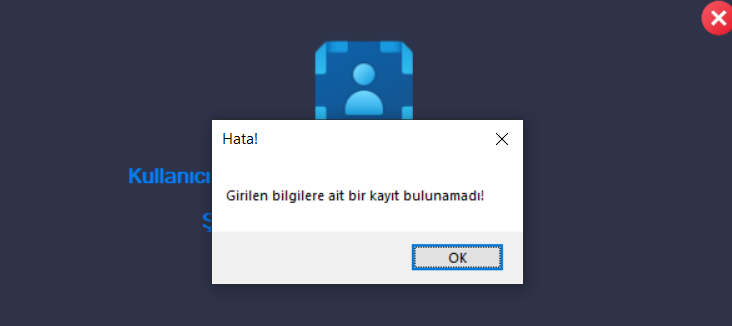
{

MessageBox.Show("Girilen bilgilere ait bir kayıt bulunamadı!","Hata!");

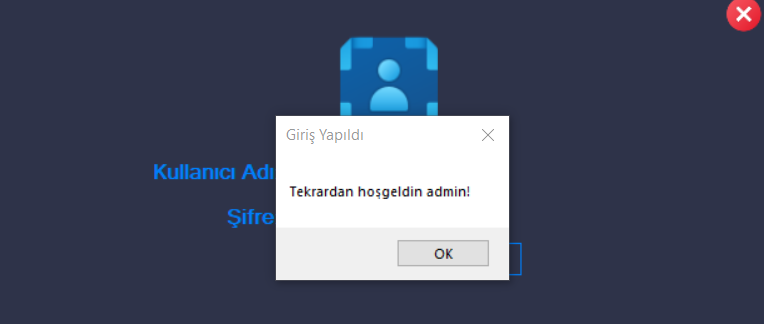
}

}

}



**Şekil 10:** Yetkili Girişi - Başarısız Giriş



**Şekil 11:** Yetkili Girişi – Başarılı Giriş

Göründüğü üzere proje çalıştırıldığında test için girilen verilerin doğruluğuna göre uyarı mesajları almaktayız.

Kullanıcı adı ve şifresini veri tabanından eşleşen verilere göre çektik ve giriş yapmış olduk.

Veriler doğru ise projemiz bizi yönetim panelinin ana sayfasına aktarmaktadır.

# Ana Sayfa / Gösterge Paneli Tasarımı



**Şekil 11:** Anasayfa Tasarımı

**Form Bilgileri:**

Form ismi: anasayfa

Gösterge Paneli: button name btnPanel

Salon ekle: button name btnSalonEkle

Bilet Satış : button name btnSatis

Film Ekle: button name btnFimEkle

Seans Ekle: button name btnSeansEkle

Seans Listele: button name btnSeansListe

Satış Listele: button name btnSatisListe

Ana Sayfa : button name btnAnasayfa

Butonların sağında sayfalar arası geçiş sağlamak için Panel : panel name pnlFormYukleyici

Sol Panel : panel name panel1

Ana sayfa formumuzun tasarımsal ve diğer formları açma komutları aşağıda gösterilmiştir:

public partial class anasayfa : Form

{

SqlConnection bag = new SqlConnection("Data Source=DESKTOP-RTSGMFH;Initial Catalog=SinemaOtomasyonu;Integrated Security=True");

[DllImport("Gdi32.dll", EntryPoint = "CreateRoundRectRgn")]

private static extern IntPtr CreateRoundRectRgn

(

int nLeftRect,

int nTopRect,

int nRightRect,

int nBottomRect,

int nWidthEllipse,

int nHeightEllipse

);

public anasayfa()

{

InitializeComponent();

Region = System.Drawing.Region.FromHrgn(CreateRoundRectRgn(0, 0, Width, Height, 25, 25));

pnlNav.Height = btnPanel.Height;

pnlNav.Top = btnPanel.Top;

pnlNav.Left = btnPanel.Left;

btnPanel.BackColor = Color.FromArgb(46, 51, 73);

lblPanelBaslik.Text = "Ana Sayfa";

this.pnlFormYukleyici.Controls.Clear();

anapage yonetimPage = new anapage() { Dock = DockStyle.Fill, TopLevel = false, TopMost = true }; // formları farklı bir sekmede açmaktansa ana tasarıma gömüyoruz

yonetimPage.FormBorderStyle = FormBorderStyle.None;

this.pnlFormYukleyici.Controls.Add(yonetimPage);

yonetimPage.Show(); // formu gösteriyoruz

}

private void anasayfa\_Load(object sender, EventArgs e)

{

}

private void btnPanel\_Click(object sender, EventArgs e)

{

pnlNav.Height = btnPanel.Height;

pnlNav.Top = btnPanel.Top;

pnlNav.Left = btnPanel.Left;

btnPanel.BackColor = Color.FromArgb(46, 51, 73);

lblPanelBaslik.Text = "Yönetim Paneli";

this.pnlFormYukleyici.Controls.Clear();

anapage yonetimPage = new anapage() { Dock = DockStyle.Fill, TopLevel = false, TopMost = true }; // formları farklı bir sekmede açmaktansa ana tasarıma gömüyoruz

yonetimPage.FormBorderStyle = FormBorderStyle.None;

this.pnlFormYukleyici.Controls.Add(yonetimPage);

yonetimPage.Show();

}

private void btnSalonlar\_Click(object sender, EventArgs e)

{

pnlNav.Height = btnSatis.Height;

pnlNav.Top = btnSatis.Top;

pnlNav.Left = btnSatis.Left;

btnSatis.BackColor = Color.FromArgb(46, 51, 73);

lblPanelBaslik.Text = "Bilet Satış";

this.pnlFormYukleyici.Controls.Clear();

biletSatis biletsatis = new biletSatis() { Dock = DockStyle.Fill, TopLevel = false, TopMost = true }; // formları farklı bir sekmede açmaktansa ana tasarıma gömüyoruz

biletsatis.FormBorderStyle = FormBorderStyle.None;

this.pnlFormYukleyici.Controls.Add(biletsatis);

biletsatis.Show();

}

private void btnFilmler\_Click(object sender, EventArgs e)

{

pnlNav.Height = btnSalonEkle.Height;

pnlNav.Top = btnSalonEkle.Top;

pnlNav.Left = btnSalonEkle.Left;

this.pnlFormYukleyici.Controls.Clear();

btnSalonEkle.BackColor = Color.FromArgb(46, 51, 73);

lblPanelBaslik.Text = "Salon Ekle";

this.pnlFormYukleyici.Controls.Clear();

salonekle salonekle = new salonekle() { Dock = DockStyle.Fill, TopLevel = false, TopMost = true }; // formları farklı bir sekmede açmaktansa ana tasarıma gömüyoruz

salonekle.FormBorderStyle = FormBorderStyle.None;

this.pnlFormYukleyici.Controls.Add(salonekle);

salonekle.Show();

}

private void btnPanel\_Leave(object sender, EventArgs e)

{

btnPanel.BackColor = Color.FromArgb(24, 30, 54);

}

private void btnFilmler\_Leave(object sender, EventArgs e)

{

btnSalonEkle.BackColor = Color.FromArgb(24, 30, 54);

}

private void btnSalonlar\_Leave(object sender, EventArgs e)

{

btnSatis.BackColor = Color.FromArgb(24, 30, 54);

}

private void btnAyarlar\_Click(object sender, EventArgs e)

{

}

private void btnSeansEkle\_Leave(object sender, EventArgs e)

{

btnSeansEkle.BackColor = Color.FromArgb(24, 30, 54);

}

private void btnFilmEkle\_Leave(object sender, EventArgs e)

{

btnFilmEkle.BackColor = Color.FromArgb(24, 30, 54);

}

private void btnSeansListe\_Leave(object sender, EventArgs e)

{

btnSeansListe.BackColor = Color.FromArgb(24, 30, 54);

}

private void btnAyarlar\_Leave(object sender, EventArgs e)

{

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

private void textBox1\_Click(object sender, EventArgs e)

{

textBox1.Clear();

}

private void label3\_Click(object sender, EventArgs e)

{

}

private void btnAnasayfa\_Click(object sender, EventArgs e)

{

lblPanelBaslik.Text = "Yönetim Paneli";

this.pnlFormYukleyici.Controls.Clear();

anapage yonetimPage = new anapage() { Dock = DockStyle.Fill, TopLevel = false, TopMost = true }; // formları farklı bir sekmede açmaktansa ana tasarıma gömüyoruz

yonetimPage.FormBorderStyle = FormBorderStyle.None;

this.pnlFormYukleyici.Controls.Add(yonetimPage);

yonetimPage.Show();

pnlNav.Height = btnPanel.Height;

pnlNav.Top = btnPanel.Top;

pnlNav.Left = btnPanel.Left;

}

private void pnlFormYukleyici\_Paint(object sender, PaintEventArgs e)

{

}

private void btnFilmEkle\_Click(object sender, EventArgs e)

{

pnlNav.Height = btnFilmEkle.Height;

pnlNav.Top = btnFilmEkle.Top;

pnlNav.Left = btnFilmEkle.Left;

btnFilmEkle.BackColor = Color.FromArgb(46, 51, 73);

lblPanelBaslik.Text = "Film Ekle";

this.pnlFormYukleyici.Controls.Clear();

Filmekle fimekle = new Filmekle() { Dock = DockStyle.Fill, TopLevel = false, TopMost = true }; // formları farklı bir sekmede açmaktansa ana tasarıma gömüyoruz

fimekle.FormBorderStyle = FormBorderStyle.None;

this.pnlFormYukleyici.Controls.Add(fimekle);

fimekle.Show();

}

private void btnSeansEkle\_Click(object sender, EventArgs e)

{

pnlNav.Height = btnSeansEkle.Height;

pnlNav.Top = btnSeansEkle.Top;

pnlNav.Left = btnSeansEkle.Left;

btnSeansEkle.BackColor = Color.FromArgb(46, 51, 73);

lblPanelBaslik.Text = "Seans Ekle";

this.pnlFormYukleyici.Controls.Clear();

Seansekle seansekle = new Seansekle() { Dock = DockStyle.Fill, TopLevel = false, TopMost = true }; // formları farklı bir sekmede açmaktansa ana tasarıma gömüyoruz

seansekle.FormBorderStyle = FormBorderStyle.None;

this.pnlFormYukleyici.Controls.Add(seansekle);

seansekle.Show();

}

private void btnSeansListe\_Click(object sender, EventArgs e)

{

pnlNav.Height = btnSeansListe.Height;

pnlNav.Top = btnSeansListe.Top;

pnlNav.Left = btnSeansListe.Left;

btnSeansListe.BackColor = Color.FromArgb(46, 51, 73);

lblPanelBaslik.Text = "Seans Listesi";

this.pnlFormYukleyici.Controls.Clear();

seansliste seansliste = new seansliste() { Dock = DockStyle.Fill, TopLevel = false, TopMost = true }; // formları farklı bir sekmede açmaktansa ana tasarıma gömüyoruz

seansliste.FormBorderStyle = FormBorderStyle.None;

this.pnlFormYukleyici.Controls.Add(seansliste);

seansliste.Show();

}

private void button2\_Click(object sender, EventArgs e)

{

pnlNav.Height = btnSeansListe.Height;

pnlNav.Top = btnSatisListe.Top;

pnlNav.Left = btnSatisListe.Left;

btnSatisListe.BackColor = Color.FromArgb(46, 51, 73);

lblPanelBaslik.Text = "Satış Listesi";

this.pnlFormYukleyici.Controls.Clear();

Satisliste satisliste = new Satisliste() { Dock = DockStyle.Fill, TopLevel = false, TopMost = true };// formları farklı bir sekmede açmaktansa ana tasarıma gömüyoru

satisliste.FormBorderStyle = FormBorderStyle.None;

this.pnlFormYukleyici.Controls.Add(satisliste);

satisliste.Show();

}

Kodlardan da göreceğiniz gibi tasarımsal açıdan kullanıcıyı yormamak adına diğer formlarımız anasayfa formumuzun üzerine gömülü olarak geliyor ve tasarımsal açıdan gözümüzü yormuyor.

Yaptığımız kullanıcı dostu bu tasarımla beraber kolayca yapılacak işlemleri seçebiliyoruz.

Öncelikle temiz ve anlaşılır bir ana sayfa / gösterge paneli oluşturduk. Buradan veri tabanımızdaki kayıtların sayısını takip edebiliyoruz.



**Şekil 12:** Anasayfa Tasarımı 2

**Form Bilgileri:**

Form ismi: anapage

Güncel Film Paneli : panel name panel3

Satış Sayısı Paneli : panel name panel4

Salon Sayısı Paneli : panel name panel5

Seans Sayısı Paneli : panel name panel6

Güncel Film Sayısının altındaki değer metnimiz : label name label4

Satış Sayısının altındaki değer metnimiz : label name label1

Salon Sayısının altındaki değer metnimiz : label name label10

Seans Sayısının altındaki değer metnimiz : label name label13

Ana sayfada yetkili personelin bilgileri güncel olarak kolayca takip edebilmesini sağladık.

Bu sayfaya güncel verileri getirmek için yazdığımız kod satırları aşağıda bulunmaktadır.

public partial class anapage : Form

{

SqlConnection bag = new SqlConnection("Data Source=DESKTOP-RTSGMFH;Initial Catalog=SinemaOtomasyonu;Integrated Security=True");

public anapage()

{

InitializeComponent();

}

int filmSayisi = -1;

int salonSayisi = -1;

int seansSayisi = -1;

int satisSayisi = -1;

private void YonetimPaneli\_Load(object sender, EventArgs e)

{

bag.Open();

SqlCommand komut = new SqlCommand("select count(\*) from film\_bilgiler",bag); // film\_bilgiler tablosundan güncel satır sayısını aldık

SqlCommand komut2 = new SqlCommand("select count(\*) from salon\_bilgileri", bag); //salon\_bilgileri tablosundan güncel satır sayısını aldık

SqlCommand komut3 = new SqlCommand("select count(\*) from seans\_bilgileri", bag); // seans\_bilgiler tablosundan güncel satır sayısını aldık

SqlCommand komut4 = new SqlCommand("select count(\*) from satis\_bilgileri", bag); // satis\_bilgiler tablosundan güncel satır sayısını aldık

filmSayisi = Convert.ToInt32(komut.ExecuteScalar());

salonSayisi = Convert.ToInt32(komut2.ExecuteScalar());

seansSayisi = Convert.ToInt32(komut3.ExecuteScalar());

satisSayisi = Convert.ToInt32(komut4.ExecuteScalar());

label4.Text = filmSayisi.ToString();

label10.Text = salonSayisi.ToString();

label13.Text = seansSayisi.ToString();

label1.Text = satisSayisi.ToString();

bag.Close();}

# Salon Ekle Sayfa Tasarımı



**Şekil 13:** Salon Ekle Tasarımı

**Form Bilgileri:**

Form ismi: salonekle

Salon adı girişi: textbox name textBox1

Salon Ekle Butonu: button name button1

Bu sayfa sinemadaki salonlara yeni bir salon eklemek için tasarlanmıştır. TextBox’a girilen veri ile yeni bir salon ekleme işlemi gerçekleştirilebilir. Eğer girilen salon adı veri tabanında bulunuyorsa bize uyarı mesajı verecek şekilde kodlanmıştır.Salon adı veri tabanında primary key olarak tanımlandığı için aynı veriler birden fazla kez girilemez.Salon ekle butonunun kodları aşağıda gösterilmiştir.

public partial class salonekle : Form

{

SqlConnection bag = new SqlConnection("Data Source=DESKTOP-RTSGMFH;Initial Catalog=SinemaOtomasyonu;Integrated Security=True");

public salonekle()

{

InitializeComponent();

}

private void salonekle\_Load(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

bag.Open();

string salonekle = "insert into Salon\_Bilgileri(SalonAdi) values (@salonadi)";

SqlCommand komut = new SqlCommand(salonekle, bag);

if (textBox1.Text=="")

{

MessageBox.Show("Lütfen Salon Adı Giriniz!","Uyarı!");

}

else

{

try

{

komut.Parameters.AddWithValue("@salonadi", textBox1.Text);

komut.ExecuteNonQuery();

MessageBox.Show("Salon Eklendi","Kayıt");

textBox1.Clear();

}

catch (Exception)

{

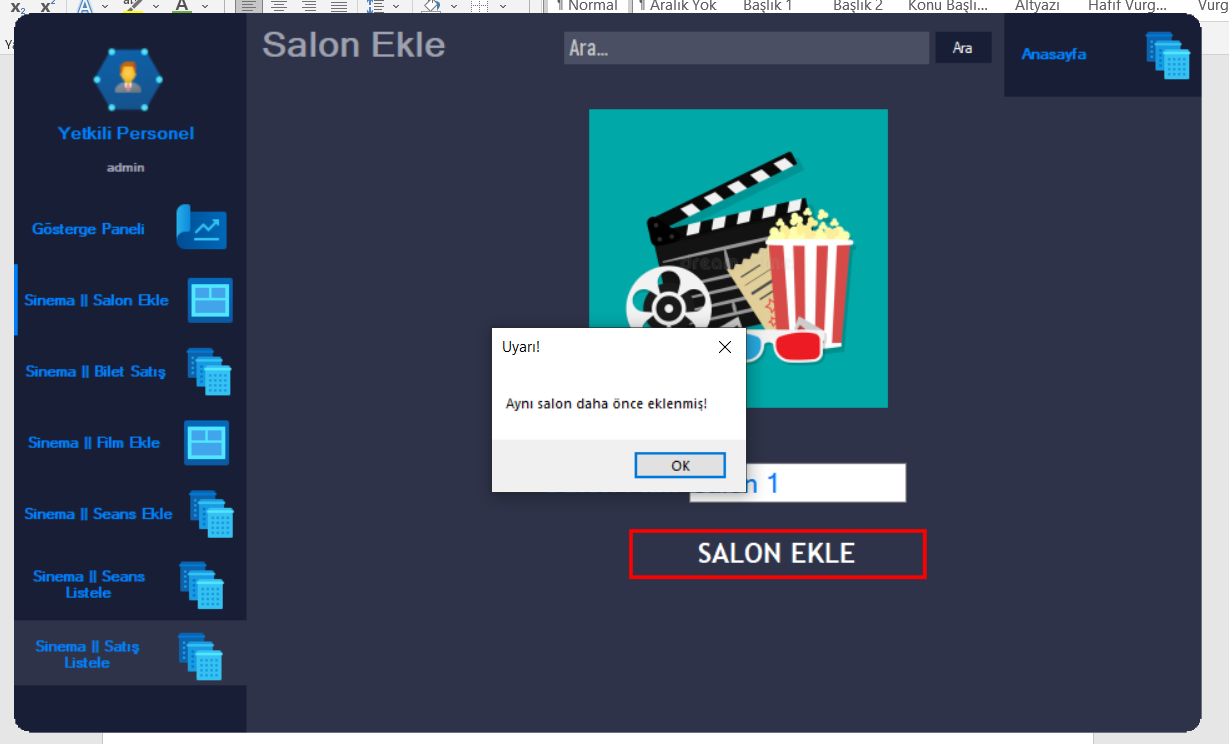
MessageBox.Show("Aynı salon daha önce eklenmiş!","Uyarı!");

}

}

bag.Close();

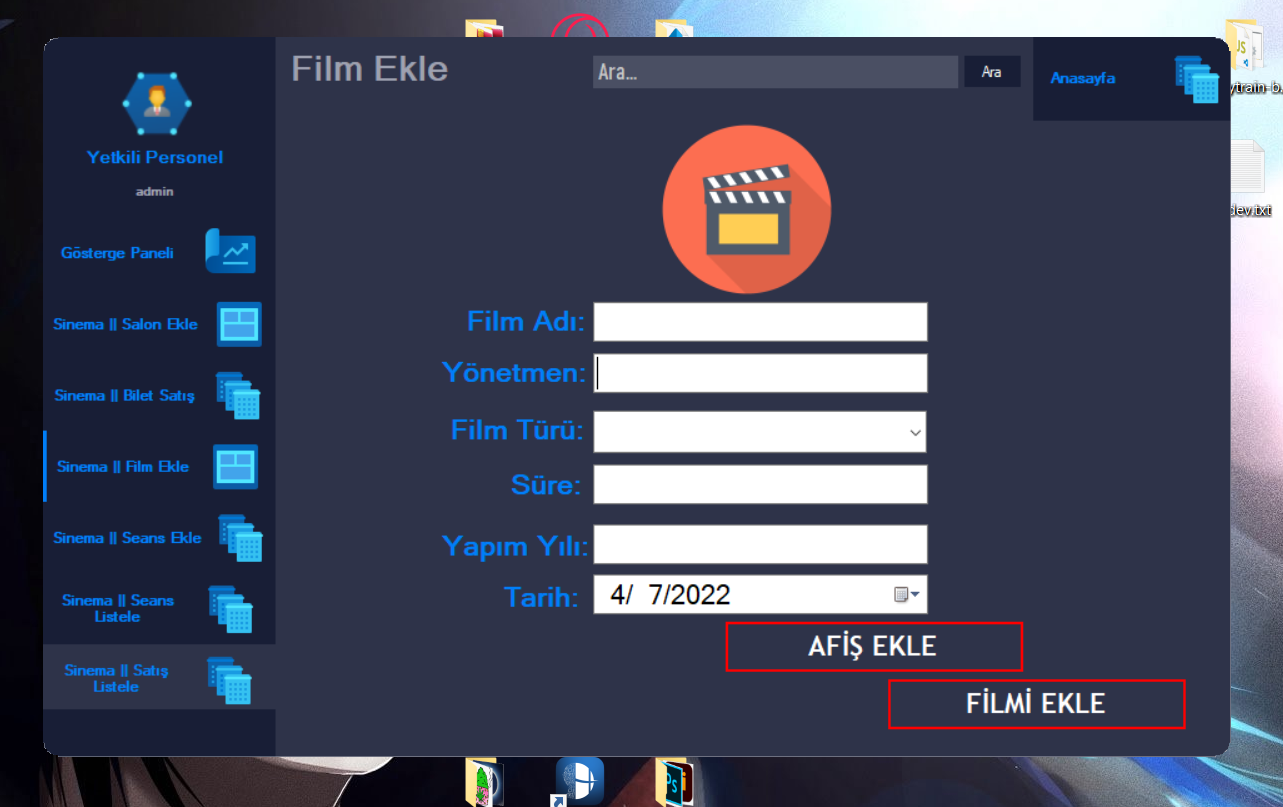
}



**Şekil 14:** Salon Ekle – Hata Mesajı

Gördüğünüz üzere önceden girilen bir salon ismi girildiğinde bize hata mesajı yönlendiriliyor.

# Film Ekle Sayfa Tasarımı



**Şekil 15:** Film Ekle Tasarımı

**Form Bilgileri:**

Film adı girişi : textBox name textBox1

Yönetmen girişi : textBox name textBox2

Film Türü seçimi : comboBox name comboBox1

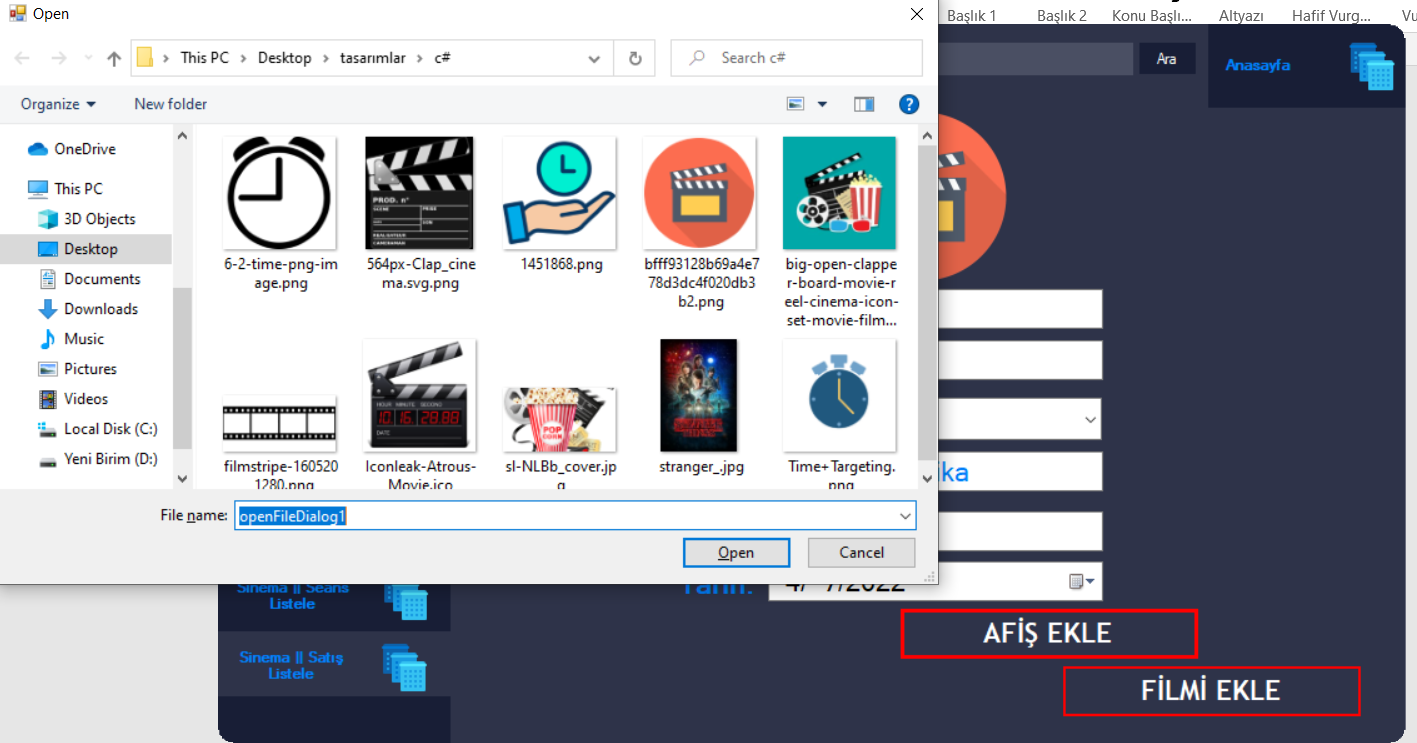
Süre girişi : textBox name textBox4

Yapım Yılı girişi : textBox name textBox5

Vizyon Tarihi Seçimi : dateTimePicker name dateTimePicker1

Afiş Görseli : pictureBox name pictureBox2

Afiş Ekle butonuna basıldığında görüldüğü üzere bizden bir film afişi seçilmesini istiyor.

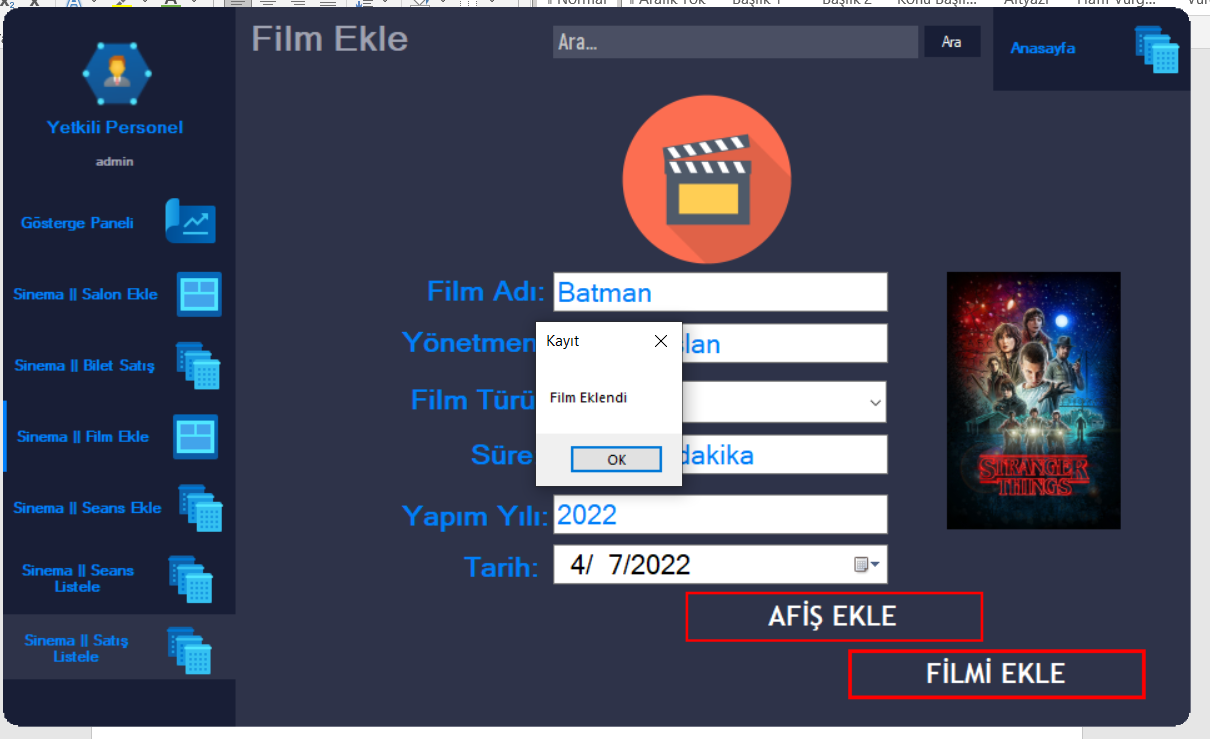


**Şekil 16:** Film Ekle – Afiş Seçimi

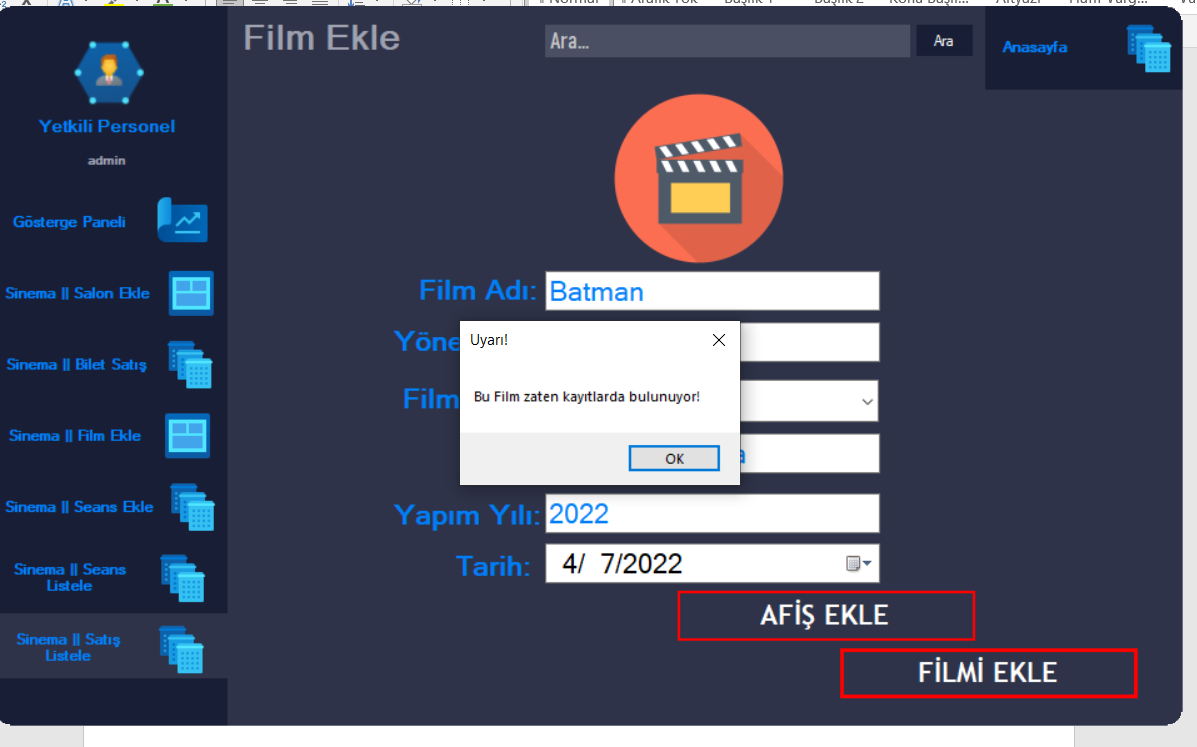


**Şekil 17:** Film Ekle – Veri Girişi

Afiş eklendiğinde ise arayüzümüze görsel olarak geliyor. Ardından Filmi Ekle butonuna basarak veri tabanına film ekleme işlemi tamamlanıyor.



**Şekil 17:** Film Ekle – Onay Mesajı



**Şekil 18:** Film Ekle –Hata Mesajı

Bu sayfaya ait kodlar aşağıda verilmiştir:

public partial class Filmekle : Form

{

SqlConnection bag = new SqlConnection("Data Source=DESKTOP-RTSGMFH;Initial Catalog=SinemaOtomasyonu;Integrated Security=True");

public Filmekle()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

bag.Open();

string filmekle = "insert into Film\_Bilgiler(FilmAdi,Yonetmen,FilmTuru,FilmSuresi,tarih,YapimYili,Resim) values (@filmadi,@yonetmen,@filmturu,@filmsuresi,@tarih,@yapimyili,@resim)";

SqlCommand komut = new SqlCommand(filmekle, bag);

try

{

komut.Parameters.AddWithValue("@filmadi", textBox1.Text);

komut.Parameters.AddWithValue("@yonetmen", textBox2.Text);

komut.Parameters.AddWithValue("@filmturu", comboBox1.Text);

komut.Parameters.AddWithValue("@filmsuresi", textBox4.Text);

komut.Parameters.AddWithValue("@yapimyili", textBox5.Text);

komut.Parameters.AddWithValue("@tarih", dateTimePicker1.Text);

komut.Parameters.AddWithValue("@resim", pictureBox2.ImageLocation);

komut.ExecuteNonQuery();

MessageBox.Show("Film Eklendi", "Kayıt");

}

catch (Exception)

{

MessageBox.Show("Bu Film zaten kayıtlarda bulunuyor!", "Uyarı!");

}

foreach (Control item in Controls) if (item is TextBox) item.Text = "";

{

comboBox1.Text = ""; // döngü ile tüm textboxları ve combobox1’i temizliyoruz

}

bag.Close();

}

private void button2\_Click(object sender, EventArgs e)

{

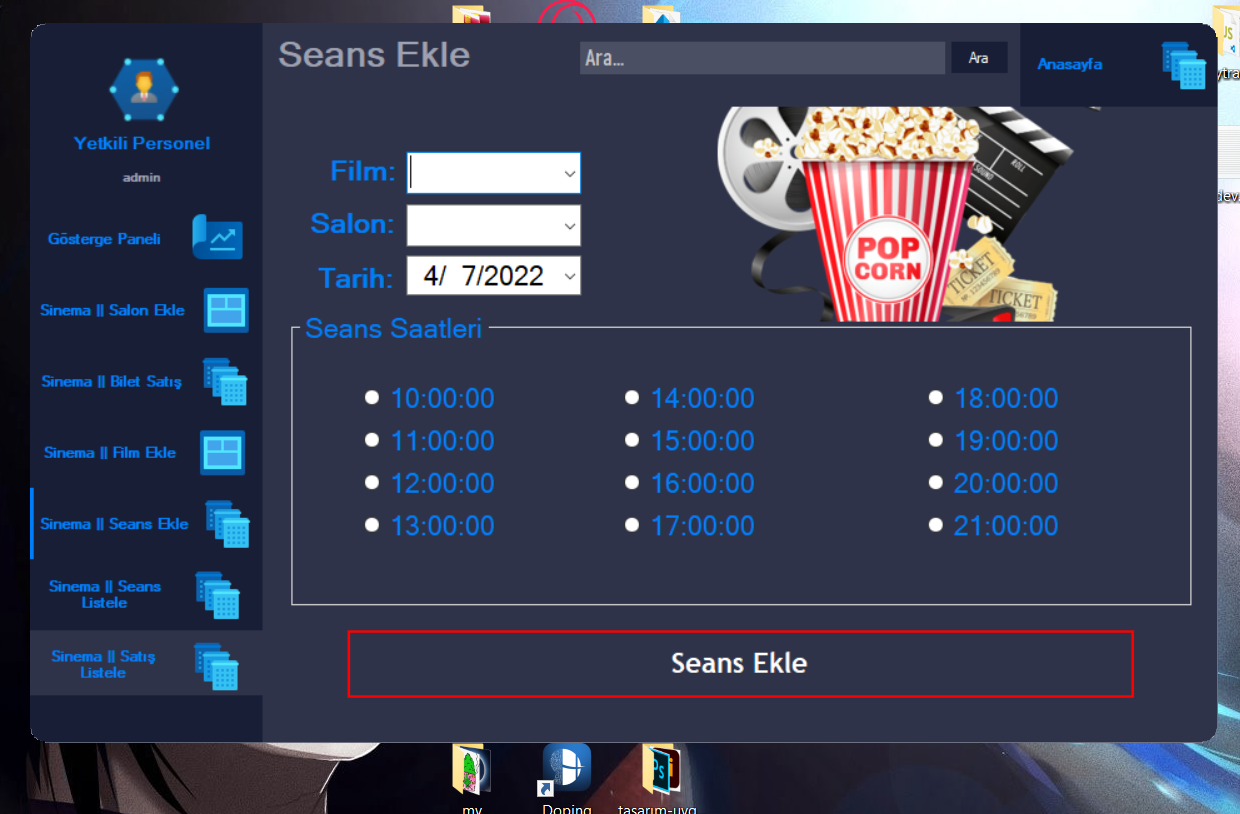
openFileDialog1.ShowDialog(); // resim seçilmesi için openfiledialog kullandık

pictureBox2.ImageLocation = openFileDialog1.FileName; //resim uzantısını openfiledialog1’de seçilen dosya ile güncelledik.

}

}

# Seans Ekle Sayfa Tasarımı



**Şekil 19:** Seans Ekle Tasarımı

**Form Bilgileri:**

Form İsmi Seansekle

Film Seçimi: combobox name comboBox1

Salon Seçimi: combobox name comboBox2

Tarih Seçimi: dateTimePicker name dateTimePicker1

Seans Saatleri: groupBox name groupBox1

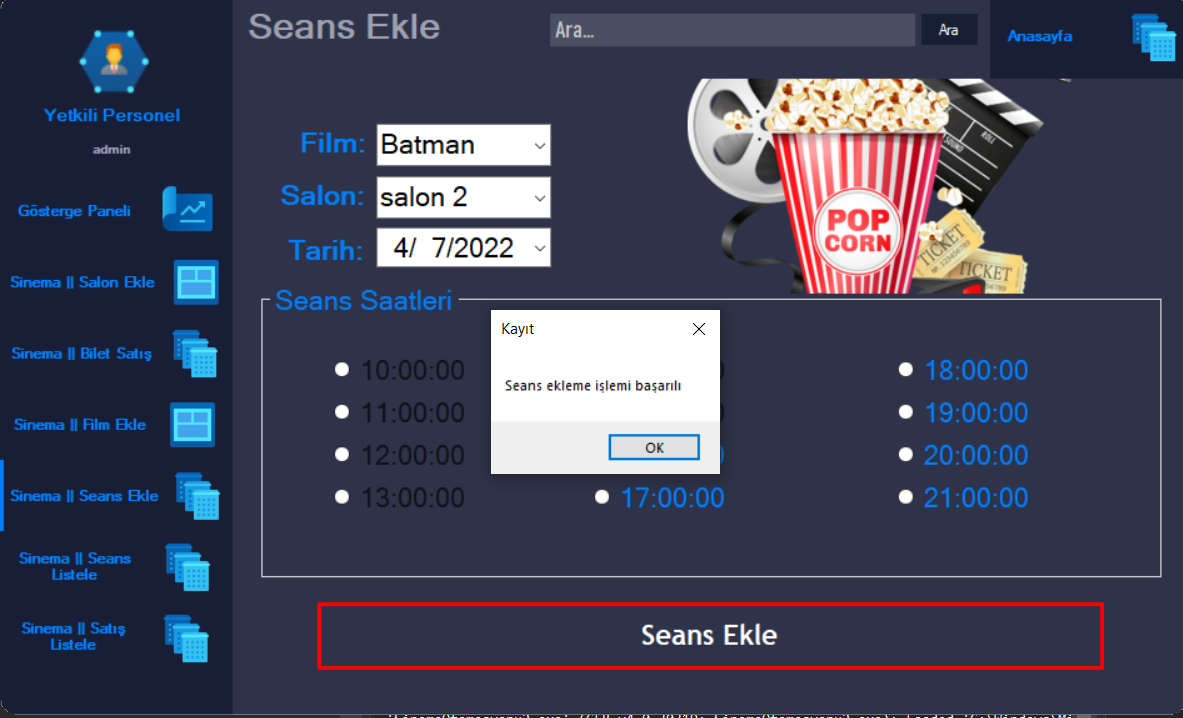
Seans saati seçimi: radioButton name radioButton1(1-12 arası toplam 12 tane farklı radioButton)

Seans Ekle: button name button1

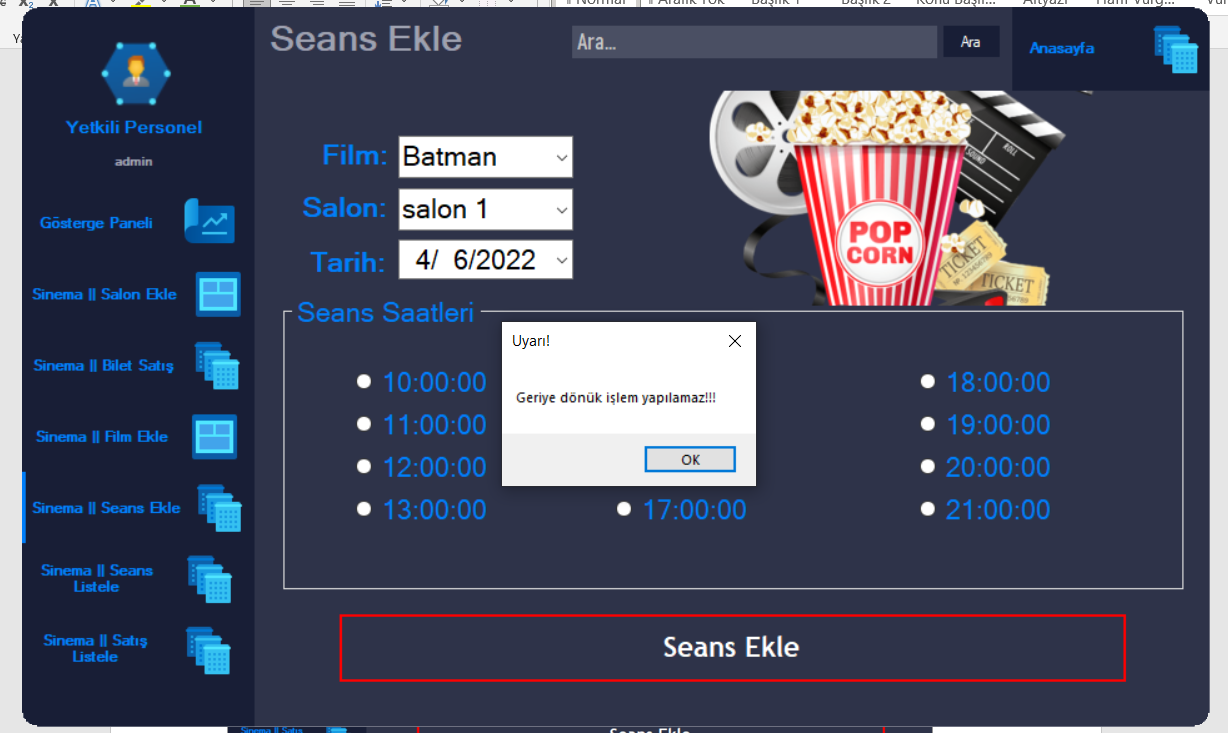
Bu sayfa seçilen film , salon ve vizyon tarihine göre seans saati seçilmesi için tasarlanmıştır.

Film ve salon bilgileri comboBox’ların içerisine otomatik olarak veri tabanından çekilerek item olarak eklenmiştir.

Geçmiş tarih ve saatlerde işlem yapılamaması için şart blokları ile bu durumun önüne geçilmiştir ve kullanıcıya uyarı kutucukları ile bildirim gönderilmektedir.



**Şekil 20:** Seans Ekle – Onay Mesajı



**Şekil 20:** Seans Ekle – Hata Mesajı

Bu sayfaya ait yazılan kodlar aşağıda gösterilmiştir:

public partial class Seansekle : Form

{

SqlConnection bag = new SqlConnection("Data Source=DESKTOP-RTSGMFH;Initial Catalog=SinemaOtomasyonu;Integrated Security=True");

public Seansekle()

{

InitializeComponent();

}

string seans = "";

private void FilmveSalonGoster(ComboBox combo,string sql,string sql2)

{

bag.Open();

SqlCommand komut = new SqlCommand(sql, bag);

SqlDataReader read = komut.ExecuteReader();

while (read.Read()==true)

{

combo.Items.Add(read[sql2].ToString());

}

bag.Close();

}

private void RadioButtonSeçiliyse()

{

if (radioButton1.Checked == true) seans = radioButton1.Text; // if / else kullanarak hangi buton işaretliyse seansa o radyo butonunun değerini atadık

else if (radioButton2.Checked == true) seans = radioButton2.Text;

else if (radioButton3.Checked == true) seans = radioButton3.Text;

else if (radioButton4.Checked == true) seans = radioButton4.Text;

else if (radioButton5.Checked == true) seans = radioButton5.Text;

else if (radioButton6.Checked == true) seans = radioButton6.Text;

else if (radioButton7.Checked == true) seans = radioButton7.Text;

else if (radioButton8.Checked == true) seans = radioButton8.Text;

else if (radioButton9.Checked == true) seans = radioButton9.Text;

else if (radioButton10.Checked == true) seans = radioButton10.Text;

else if (radioButton11.Checked == true) seans = radioButton11.Text;

else if (radioButton12.Checked == true) seans = radioButton12.Text;

}

private void Seansekle\_Load(object sender, EventArgs e)

{

FilmveSalonGoster(comboBox1, "select \* from film\_bilgiler","filmadi"); // combobox1'e film isinlerini getirdik

FilmveSalonGoster(comboBox2, "select \* from salon\_bilgileri", "salonadi"); // combobox2 ye salon isimlerini getirdik

}

private void button1\_Click(object sender, EventArgs e)

{

RadioButtonSeçiliyse();

if (seans!="")

{

bag.Open();

string seansekle = "insert into Seans\_Bilgileri(FilmAdi,SalonAdi,Tarih,Seans) values (@filmadi,@salonadi,@tarih,@seans)";

SqlCommand komut = new SqlCommand(seansekle, bag);

komut.Parameters.AddWithValue("@filmadi", comboBox1.Text);

komut.Parameters.AddWithValue("@salonadi", comboBox2.Text);

komut.Parameters.AddWithValue("@tarih", dateTimePicker1.Text);

komut.Parameters.AddWithValue("@seans", seans);

komut.ExecuteNonQuery();

MessageBox.Show("Seans ekleme işlemi başarılı", "Kayıt");

bag.Close();

}

else if (seans=="")

{

MessageBox.Show("Seans seçimi yapmadınız!!!", "Uyarı!");

}

comboBox1.Text = "";

comboBox2.Text = "";

dateTimePicker1.Text = DateTime.Now.ToShortDateString();

}

private void dateTimePicker1\_ValueChanged(object sender, EventArgs e)

{

foreach (Control item3 in groupBox1.Controls)

{

item3.Enabled = true;

}

DateTime bugün = DateTime.Parse(DateTime.Now.ToShortDateString());

DateTime yeni = DateTime.Parse(dateTimePicker1.Text);

if (yeni==bugün)

{

foreach (Control item in groupBox1.Controls)

{

if (DateTime.Parse(DateTime.Now.ToShortTimeString()) > DateTime.Parse(item.Text)) // radyo butonlar şuanki saatten küçükse pasif ediliyor

{

item.Enabled = false;

}

}

Tarihi\_karsilastir();

}

else if (yeni>bugün)

{

Tarihi\_karsilastir();

}

else if (yeni<bugün)

{

MessageBox.Show("Geriye dönük işlem yapılamaz!!!","Uyarı!");

dateTimePicker1.Text = DateTime.Now.ToShortDateString();

}

}

private void Tarihi\_karsilastir()

{

bag.Open();

SqlCommand komut = new SqlCommand("select \* from seans\_bilgileri where salonadi='" + comboBox2.Text + "'and tarih='" + dateTimePicker1.Text + "'", bag);

SqlDataReader read = komut.ExecuteReader();

while (read.Read() == true)

{

foreach (Control item2 in groupBox1.Controls)

{

if (read["seans"].ToString() == item2.Text)

{

item2.Enabled = false;

}

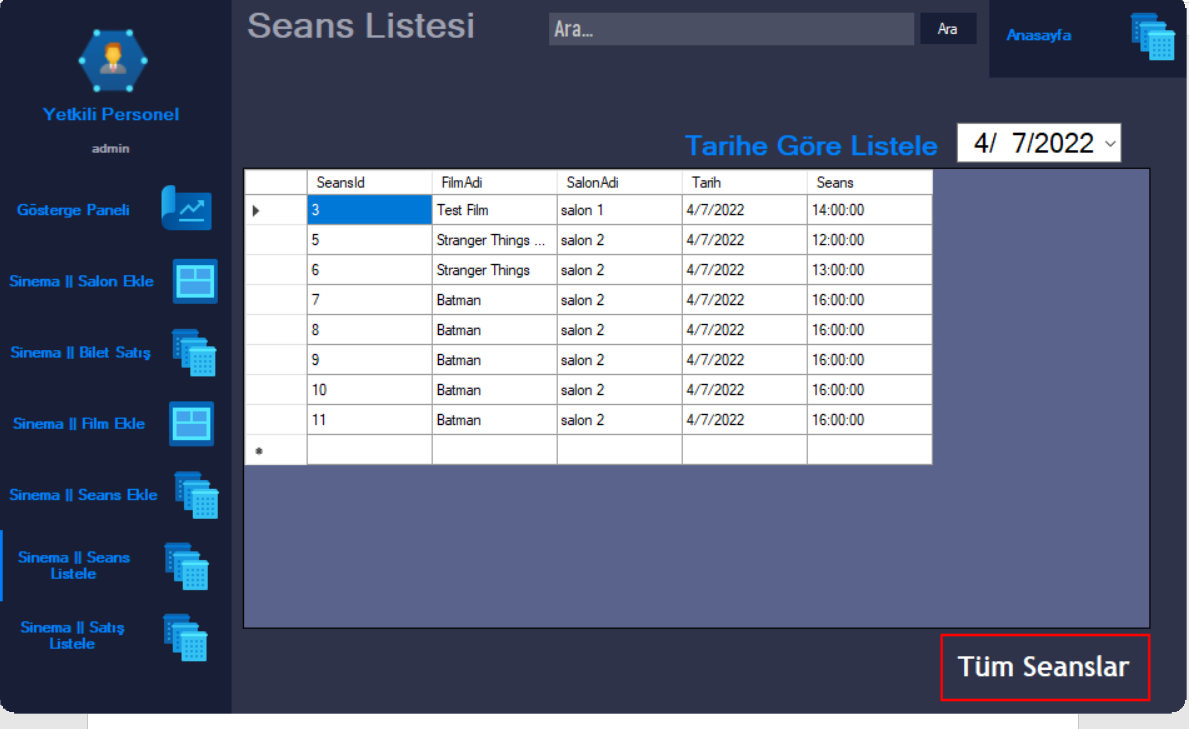
}

}

bag.Close();

}  
}

# Seans Listele Sayfa Tasarımı



**Şekil 21:** Seans Liste Tasarımı

**Form Bilgileri:**

Form ismi : seansliste

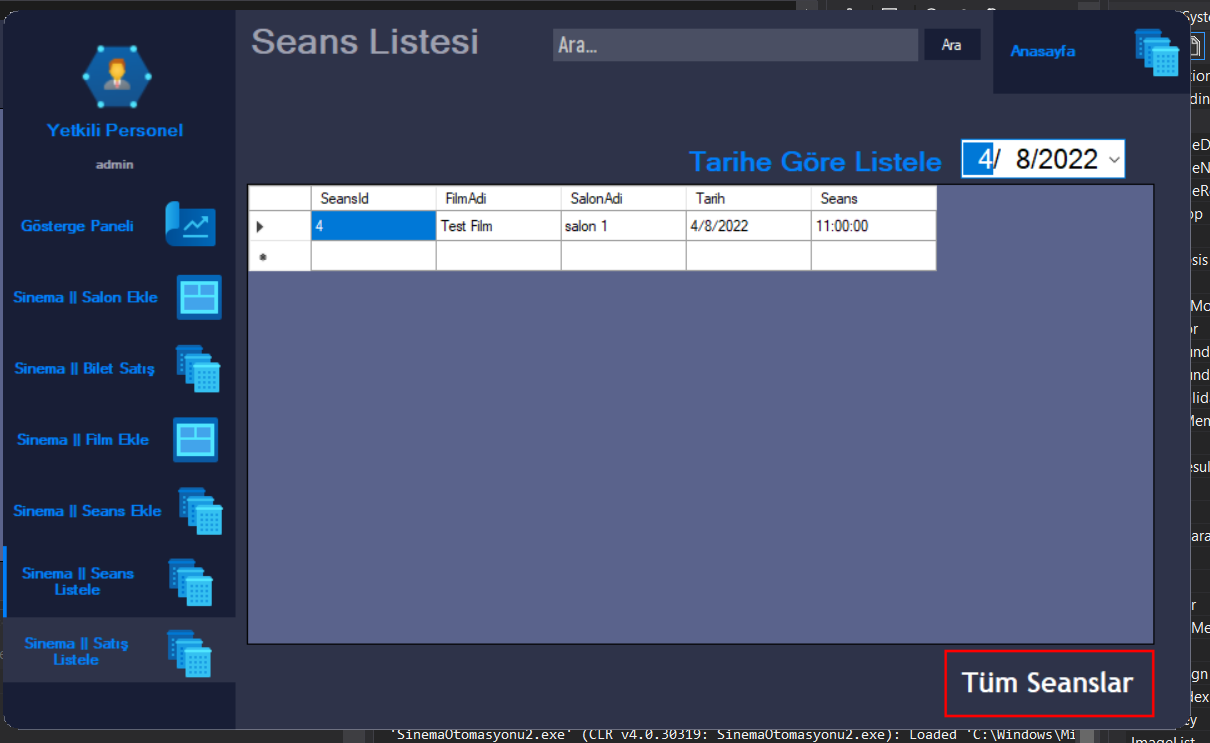
Verilerin Gösterildiği Alan: dataGridView name dataGridView1

Tarihe Göre Listeleme Opsiyonu: dateTimePicker name dateTimePicker1

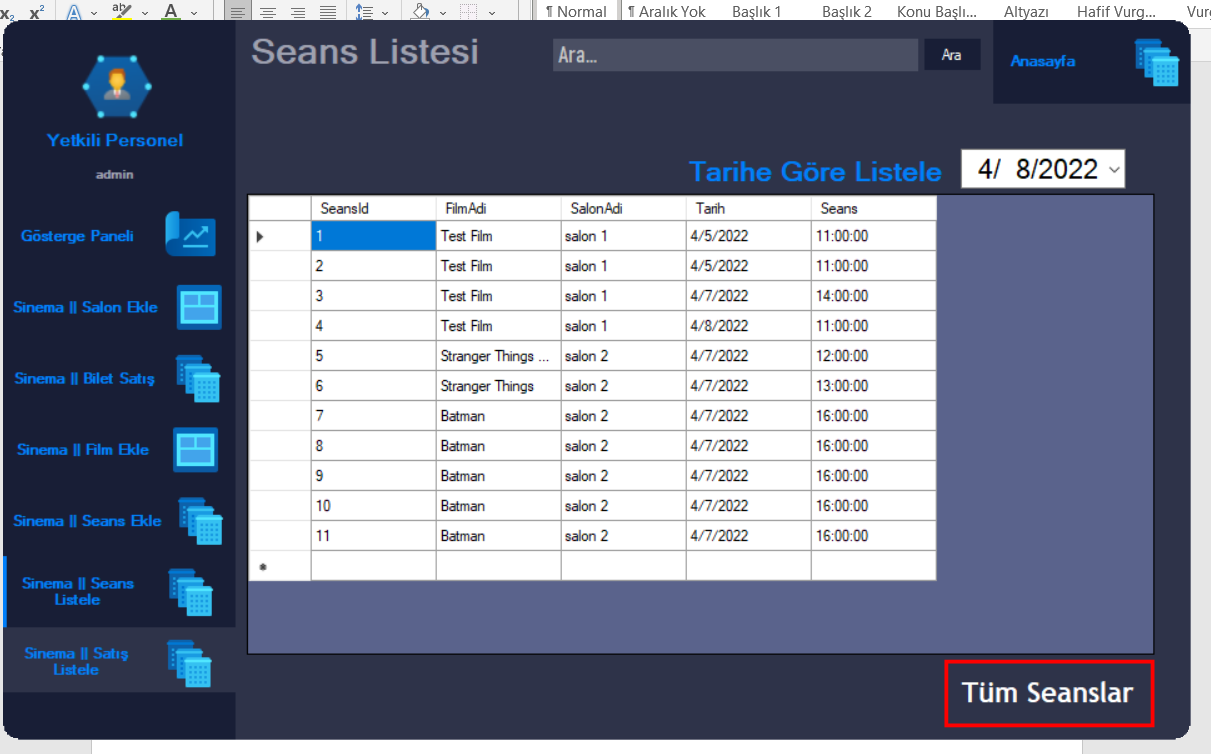
Tüm Seansları Listeleme Opsiyonu : button name button1

Bu sayfada oluşturulmuş seansları görüntülemeyi sağladık. Ek olarak hangi tarihte hangi seanlar var gibi sorunlara çözüm bulmak adına tarihe göre sıralama opsiyonu da ekledik.

Ayrıca tüm seanslara da bakma adına tüm seanslar adlı buton oluşturularak bu soruna çözüm bulduk.



**Şekil 22:** Seans Liste Tasarımı 2



**Şekil 23:** Seans Liste Tasarımı 3

Bu sayfaya ait yazdığımız kodlar aşağıda gösterilmiştir:

public partial class seansliste : Form

{

SqlConnection bag = new SqlConnection("Data Source=DESKTOP-RTSGMFH;Initial Catalog=SinemaOtomasyonu;Integrated Security=True");

DataTable tablo = new DataTable();

private void SeansListesi(string sql)

{

bag.Open();

SqlDataAdapter adtr = new SqlDataAdapter(sql, bag);

adtr.Fill(tablo);

dataGridView1.DataSource = tablo;

bag.Close();

}

public seansliste()

{

InitializeComponent();

}

private void seansliste\_Load(object sender, EventArgs e)

{

tablo.Clear();

SeansListesi("select \* from seans\_bilgileri where tarih like '" + dateTimePicker1.Text + "'");

}

private void dateTimePicker1\_ValueChanged(object sender, EventArgs e)

{

tablo.Clear();

SeansListesi("select \* from seans\_bilgileri where tarih like '" + dateTimePicker1.Text + "'");

}

private void button1\_Click(object sender, EventArgs e)

{

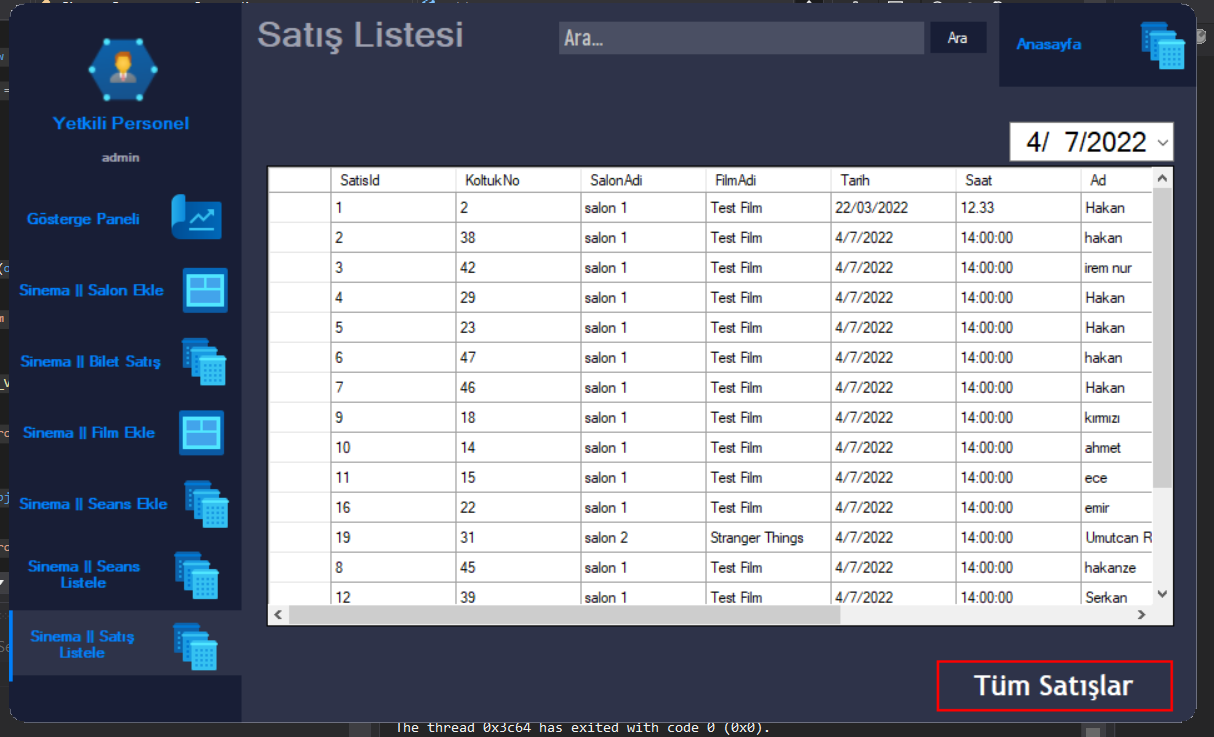
tablo.Clear();

SeansListesi("select \* from seans\_bilgileri");

}

}

# Satış Listele Sayfa Tasarımı



**Şekil 24:** Satış Listele Tasarımı

**Form Bilgileri:**

Form ismi : Satisliste

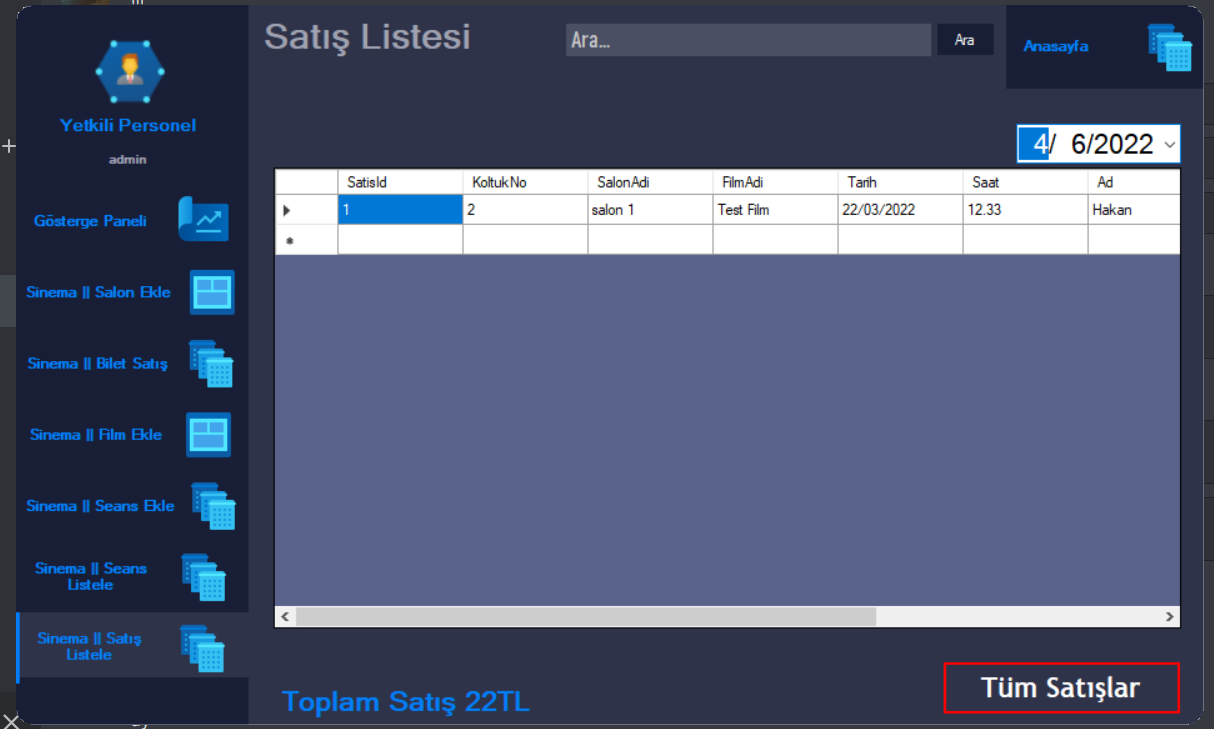
Tarihe göre listeleme opsiyonu: dateTimePicker name dateTimePicker1

Verilerin listelendiği yer: dataGridView name dataGridView1

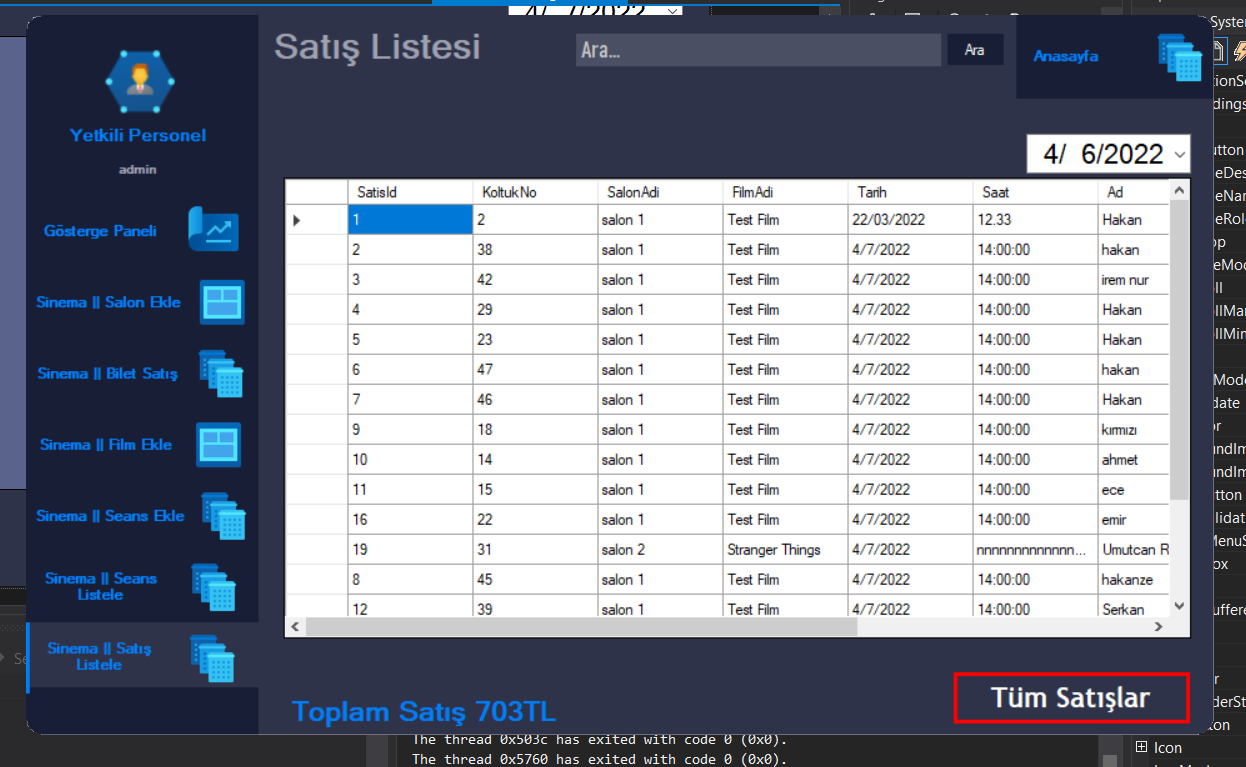
Tüm satışları listeleme opsiyonu: button name button1

Toplam satışlardan elde edilen kazancı göstermek için : label name label1

Bu sayfada personellerin yapılan satışların takibini sağlayabilmelerini istedik. Yapılan satışlarda isim ve soyisim almamızın sebebi kriminal veya herhangi bir suç teşkil eden olayda bilet sahibinin o saatte sinemada olduğunu kanıtlayabilmesini sağlamaktır. Ayrıca yapılan tüm satışlardan elde edilen toplam kazancı label üzerinden yazdırdık.



**Şekil 25:** Satış Listele Tasarımı 2



**Şekil 26:** Satış Listele Tasarımı 3

Bu sayfaya ait yazdığımız kodlar aşağıda sunulmuştur.

public partial class Satisliste : Form

{

SqlConnection bag = new SqlConnection("Data Source=DESKTOP-RTSGMFH;Initial Catalog=SinemaOtomasyonu;Integrated Security=True");

DataTable tablo = new DataTable();

public Satisliste()

{

InitializeComponent();

}

private void SatisListesi(string sql)

{

bag.Open();

SqlDataAdapter adtr = new SqlDataAdapter(sql, bag);

adtr.Fill(tablo);

dataGridView1.DataSource = tablo;

bag.Close();

}

private void Satisliste\_Load(object sender, EventArgs e)

{

tablo.Clear();

SatisListesi("select \* from satis\_bilgileri");

}

private void ToplamUcretHesapla()

{

int ucrettoplami = 0;

for (int i = 0; i < dataGridView1.Rows.Count; i++)

{

ucrettoplami += Convert.ToInt32(dataGridView1.Rows[i].Cells["ucret"].Value);

}

label1.Text = " Toplam Satış " + ucrettoplami + "TL";

}

private void button1\_Click(object sender, EventArgs e)

{

tablo.Clear();

SatisListesi("select \* from satis\_bilgileri");

ToplamUcretHesapla();

}

private void dateTimePicker1\_ValueChanged(object sender, EventArgs e)

{

tablo.Clear();

SatisListesi("select \* from satis\_bilgileri where SatisTarihi like '"+dateTimePicker1.Text +"'");

ToplamUcretHesapla();

}

}

# Bilet Satış Sayfa Tasarımı



**Şekil 27:** Bilet Satış Tasarımı

**Form Bilgileri:**

Form ismi: biletSatis

Koltuklar : groupBox name groupBox3

Bilet Satış İşlemleri: groupBox name groupBox1

Bilet İptal: groupBox name groupBox2

Film adı seçimi: comboBox name combobox1

Salon adı seçimi: comboBox name combobox2

Film tarihi seçimi: comboBox name combobox3

Film seansı seçimi: comboBox name combobox4

Koltuk No: textBox name textBox1

Ad: textbox name textBox2

Soyad: textbox name textBox3

Ücret: comboBox name comboBox8

Koltuk No iptal: comboBox name comboBox9

Bilet sat: button name button1

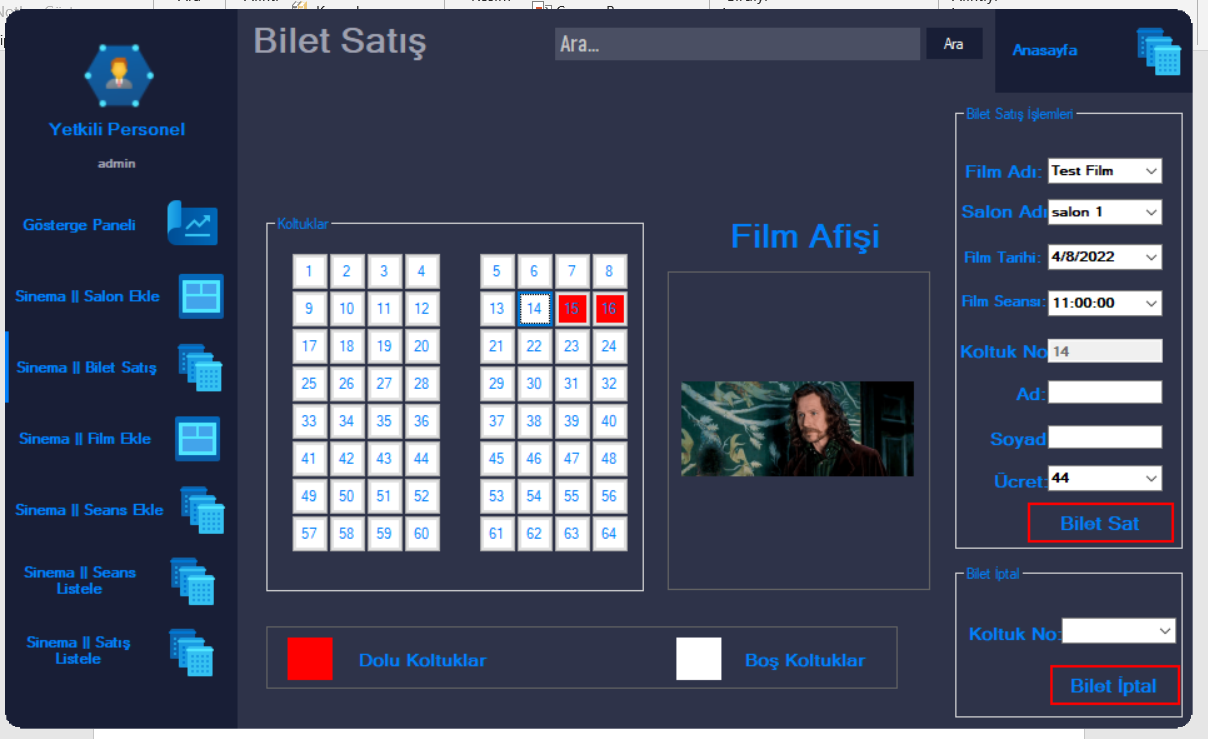
Bilet iptal: button name button2

Afiş görseli: pictureBox name pictureBox1

Koltuk Butonları: btn (işlemlerde kolaylık adına kod ile oluşturuldu)

Bu sayfada müşterilere görsel olarak istedikleri koltukları seçebilmelerini sağladık. Seçilen koltuk üzerinden kişilere bilet kesme işlemi yapmak için çeşitli opsiyonlar kullandık. Bunlar;

Film seçimi,salon seçimi,tarih seçimi,seans seçimi , koltuk seçimi ve ücret tarifesi.



**Şekil 28:** Bilet Satış Tasarımı 2

Şekilde gösterildiği üzere salon tarih ve seans bilgileri girildiği takdirde bize dolu olan koltuklar kırmızı renkte gözüküyor. Kırmızı renkli koltuklar üzerinde işlem gerçekleştiremiyoruz. Yalnızca beyaz koltuklar üzerinden satın alım yapabiliyoruz. Bu sayede yaşanabilecek teknik aksaklıkların önüne geçmiş olduk.

Film seçildiği takdirde film afişi otomatik olarak önümüze sunuldu.  
Hatalı kullanımın önüne geçmek adına koltuk no’ya ait textbox’ın kullanılırlığı in aktif edildi. Koltuk numarasına sahip butonlara tıklandığı taktirde koltuk numarası otomatik olarak textbox’a yazdırılmaktadır.



**Şekil 29:** Bilet Satış Tasarımı 3

Bilet iptal etme opsiyonunda doğru işlem gerçekleştirmek adına comboBox’a sadece kırmızı renkli olan koltukları getirilmiştir. Bu sayede olası bir yanlış kullanımın önüne geçilmiştir.

Bu sayfaya ait kodlar aşağıda gösterilmiştir.

public partial class biletSatis : Form

{

SqlConnection bag = new SqlConnection("Data Source=DESKTOP-RTSGMFH;Initial Catalog=SinemaOtomasyonu;Integrated Security=True");

public biletSatis()

{

InitializeComponent();

}

int sayac = 0;

private void FilmveSalonGetir(ComboBox combo,string sql1,string sql2)

{

bag.Open();

SqlCommand komut = new SqlCommand(sql1,bag);

SqlDataReader read = komut.ExecuteReader();

while (read.Read())

{

combo.Items.Add(read[sql2].ToString());

}

bag.Close();

}

private void combo\_dolu\_koltuklar()

{

comboBox9.Items.Clear();

comboBox9.Text = "";

foreach (Control item in groupBox3.Controls)

{

if (item is Button)

{

if (item.BackColor==Color.Red)

{

comboBox9.Items.Add(item.Text);

}

}

}

}

private void vt\_dolu\_koltuklar()

{

bag.Open();

SqlCommand komut = new SqlCommand("select \* from satis\_bilgileri where filmadi='"+comboBox1.Text+"' and salonadi= '"+comboBox2.Text+"'and tarih = '"+comboBox3.SelectedItem+"'and saat='"+comboBox4.SelectedItem+"'",bag);

SqlDataReader reader = komut.ExecuteReader();

while (reader.Read())

{

foreach (Control item in groupBox3.Controls)

{

if (item is Button)

{

if (reader["koltukno"].ToString()==item.Text)

{

item.BackColor = Color.Red;

}

}

}

}

bag.Close();

}

private void yenidenRenklendir()

{

foreach (Control item in groupBox3.Controls)

{

if (item is Button)

{

item.BackColor = Color.White;

}

}

}

private void FilmAfisiGoster()

{ bag.Open();

SqlCommand komut = new SqlCommand("select \* from film\_bilgiler where filmadi='"+comboBox1.SelectedItem+"'", bag);

SqlDataReader read = komut.ExecuteReader();

while (read.Read())

{

pictureBox1.ImageLocation = read["resim"].ToString();

}

bag.Close();

}

private void biletSatis\_Load(object sender, EventArgs e)

{

Bos\_Koltuklar();

FilmveSalonGetir(comboBox1,"select\*from film\_bilgiler","filmadi");

FilmveSalonGetir(comboBox2, "select\*from salon\_bilgileri", "salonadi");

}

private void Bos\_Koltuklar()

{

sayac = 1;

for (int i = 0; i < 8; i++)

{

for (int j = 0; j < 9; j++)

{

Button btn = new Button();

btn.Size = new Size(30, 30);

btn.BackColor = Color.White;

btn.Location = new Point(j \* 30 + 20, i \* 30 + 30);

btn.Name = sayac.ToString();

btn.Text = sayac.ToString();

if (j == 4)

{

continue;

}

sayac++;

this.groupBox3.Controls.Add(btn);

btn.Click += Btn\_Click;

}

}

}

private void Btn\_Click(object sender, EventArgs e)

{

Button b = (Button)sender;

if (b.BackColor==Color.White)

{

textBox1.Text = b.Text;

}

}

private void button1\_Click(object sender, EventArgs e)

{

bag.Open();

string filmekle = "insert into Satis\_Bilgileri(KoltukNo,SalonAdi,FilmAdi,Tarih,Saat,Ad,Soyad,Ucret,SatisTarihi) values (@koltukno,@salonadi,@filmadi,@filmtarihi,@saat,@ad,@soyad,@ucret,@satistarihi)";

SqlCommand komut = new SqlCommand(filmekle, bag);

if (textBox1.Text != "")

{

try

{

komut.Parameters.AddWithValue("@filmadi", comboBox1.Text);

komut.Parameters.AddWithValue("@salonadi", comboBox2.Text);

komut.Parameters.AddWithValue("@filmtarihi", comboBox3.Text);

komut.Parameters.AddWithValue("@saat", comboBox4.Text);

komut.Parameters.AddWithValue("@koltukno", textBox1.Text);

komut.Parameters.AddWithValue("@ad", textBox2.Text);

komut.Parameters.AddWithValue("@soyad", textBox3.Text);

komut.Parameters.AddWithValue("@ucret", comboBox8.Text);

komut.Parameters.AddWithValue("@satistarihi", DateTime.Now.ToString());

komut.ExecuteNonQuery();

foreach (Control item in groupBox1.Controls) if (item is TextBox) item.Text = "";

MessageBox.Show("Bilet Kesildi", "Kayıt");

bag.Close();

yenidenRenklendir();

vt\_dolu\_koltuklar();

combo\_dolu\_koltuklar();

}

catch (Exception)

{

MessageBox.Show("hata", "hata");

}

}

else

{

MessageBox.Show("Koltuk seçimi yapmadınız!!!", "Uyarı!");

}

bag.Close();

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

comboBox4.Items.Clear();

comboBox3.Items.Clear();

comboBox4.Text = "";

comboBox3.Text = "";

foreach (Control item in groupBox3.Controls) if (item is TextBox) item.Text = "";

FilmAfisiGoster();

yenidenRenklendir();

combo\_dolu\_koltuklar();

}

private void comboBox9\_SelectedIndexChanged(object sender, EventArgs e)

{

}

private void Film\_Tarihi\_getir()

{

bag.Open();

SqlCommand komut = new SqlCommand("select \* from seans\_bilgileri where filmadi='"+comboBox1.SelectedItem+"' and salonadi='"+comboBox2.SelectedItem+"'",bag);

SqlDataReader read = komut.ExecuteReader();

while (read.Read())

{

if (DateTime.Parse(read["tarih"].ToString())>=DateTime.Parse(DateTime.Now.ToShortDateString()))

{

if (!comboBox3.Items.Contains(read["tarih"].ToString()))

{

comboBox3.Items.Add(read["tarih"].ToString());

}

}

}

bag.Close();

}

private void comboBox2\_SelectedIndexChanged(object sender, EventArgs e)

{

Film\_Tarihi\_getir();

}

private void Film\_Seansi\_Getir() {

comboBox4.Text = "";

comboBox4.Items.Clear();

bag.Open();

SqlCommand komut = new SqlCommand("select \* from seans\_bilgileri where filmadi='" + comboBox1.SelectedItem + "' and salonadi='" + comboBox2.SelectedItem + "'and tarih='"+comboBox3.SelectedItem+"'", bag);

SqlDataReader read = komut.ExecuteReader();

while (read.Read())

{

if (DateTime.Parse(read["tarih"].ToString()) == DateTime.Parse(DateTime.Now.ToShortDateString()))

{

if (DateTime.Parse(read["seans"].ToString())>DateTime.Parse(DateTime.Now.ToShortTimeString()))

{

comboBox4.Items.Add(read["seans"].ToString());

}

}

else if (DateTime.Parse(read["tarih"].ToString()) > DateTime.Parse(DateTime.Now.ToShortDateString()))

{

comboBox4.Items.Add(read["seans"].ToString());

}

}

bag.Close();

}

private void comboBox3\_SelectedIndexChanged(object sender, EventArgs e)

{

Film\_Seansi\_Getir();

}

private void comboBox4\_SelectedIndexChanged(object sender, EventArgs e)

{

yenidenRenklendir();

vt\_dolu\_koltuklar();

combo\_dolu\_koltuklar();

}

private void button2\_Click(object sender, EventArgs e)

{

bag.Open();

if (comboBox9.Text!="")

{

try

{

SqlCommand verisil = new SqlCommand("delete from Satis\_Bilgileri where filmadi='" + comboBox1.Text + "'and salonadi='" + comboBox2.Text + "'and tarih='" + comboBox3.SelectedItem + "'and saat='" + comboBox4.Text + "'and koltukno='" + comboBox9.Text + "'",bag);

verisil.ExecuteNonQuery();

bag.Close();

yenidenRenklendir();

vt\_dolu\_koltuklar();

combo\_dolu\_koltuklar();

}

catch (Exception hata)

{

MessageBox.Show("Hata oluştu!"+hata.Message, "Uyarı");

}

}

else

{

MessageBox.Show("Koltuk seçimi yapmadınız!", "Uyarı!");

}

bag.Close();

}

}