

**ANKARA ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



BLM4531 PROJE RAPORU

Kütüphane Sistemi Yönetimi

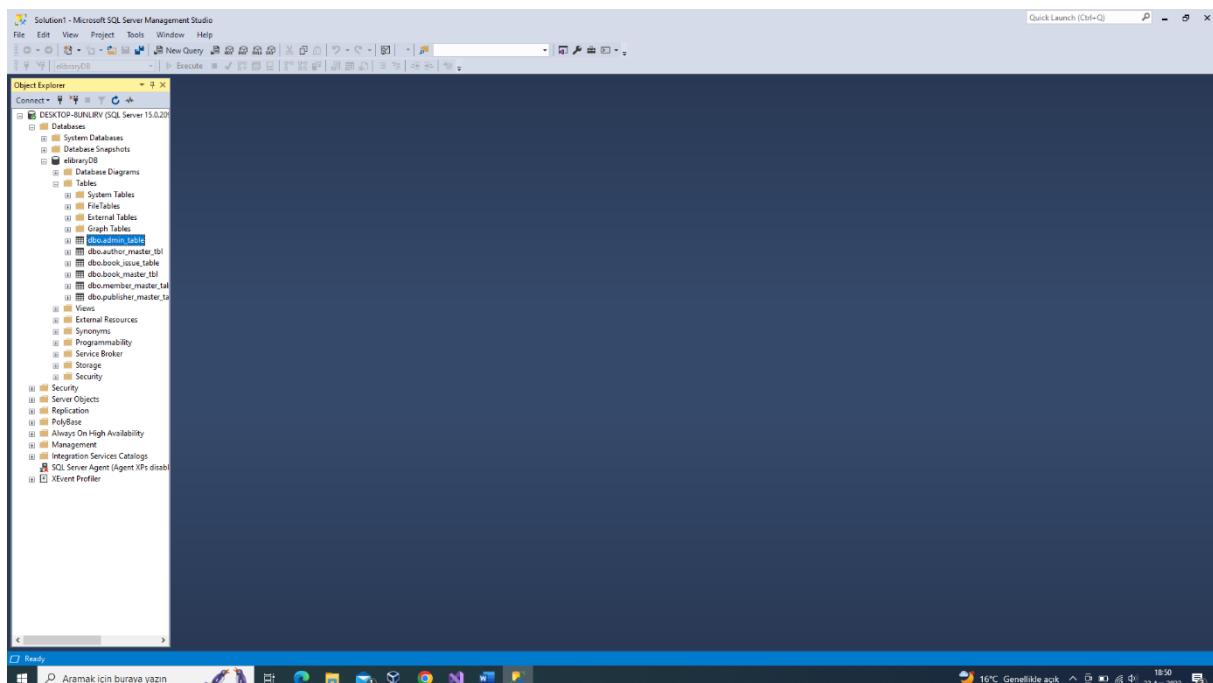
Serenay ÇELİKKAYA

18291019

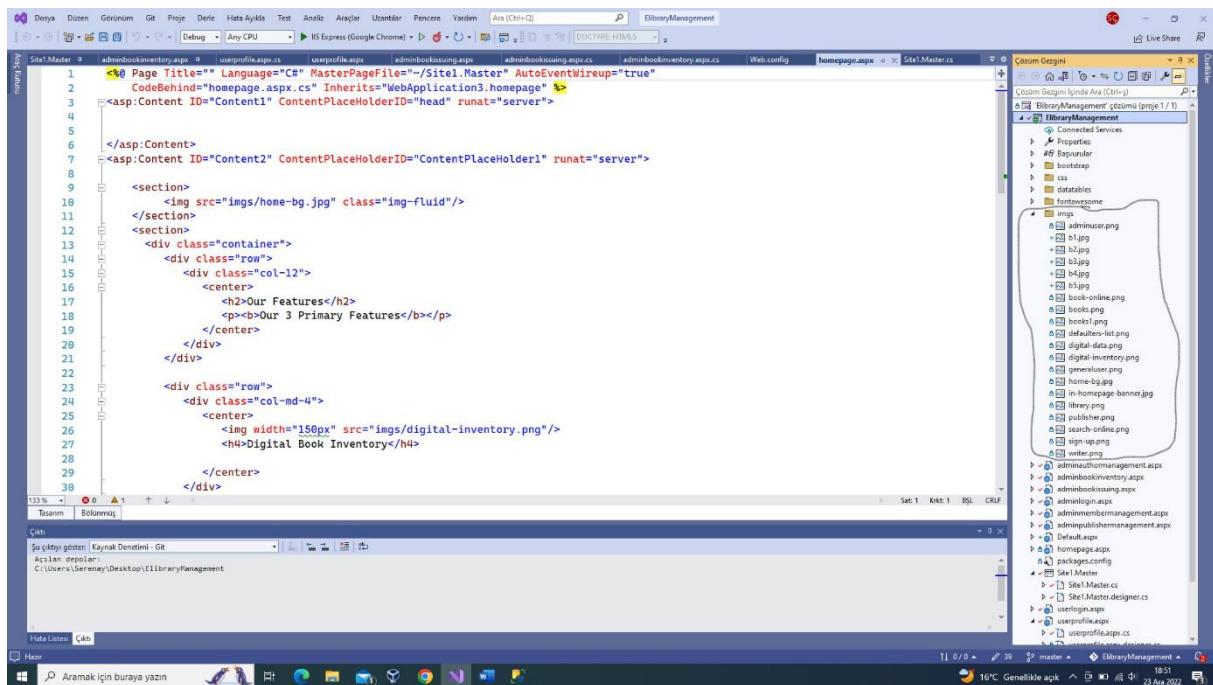
Öğr. Gör. Enver BAĞCI

Proje sunumumda backend, frontend, veritabanı ve api kısımları ayrı bölünlenmemiş olup her maddenin(1,2,3,4,5) kendi içinde bunların anlatımları yapılmıştır. Eklenen sayfa resimleri frontend, eklenen kodlar ve açıklamaları backend ve api, eklenen tablo resimleri veri tabanı kısmına aittir.

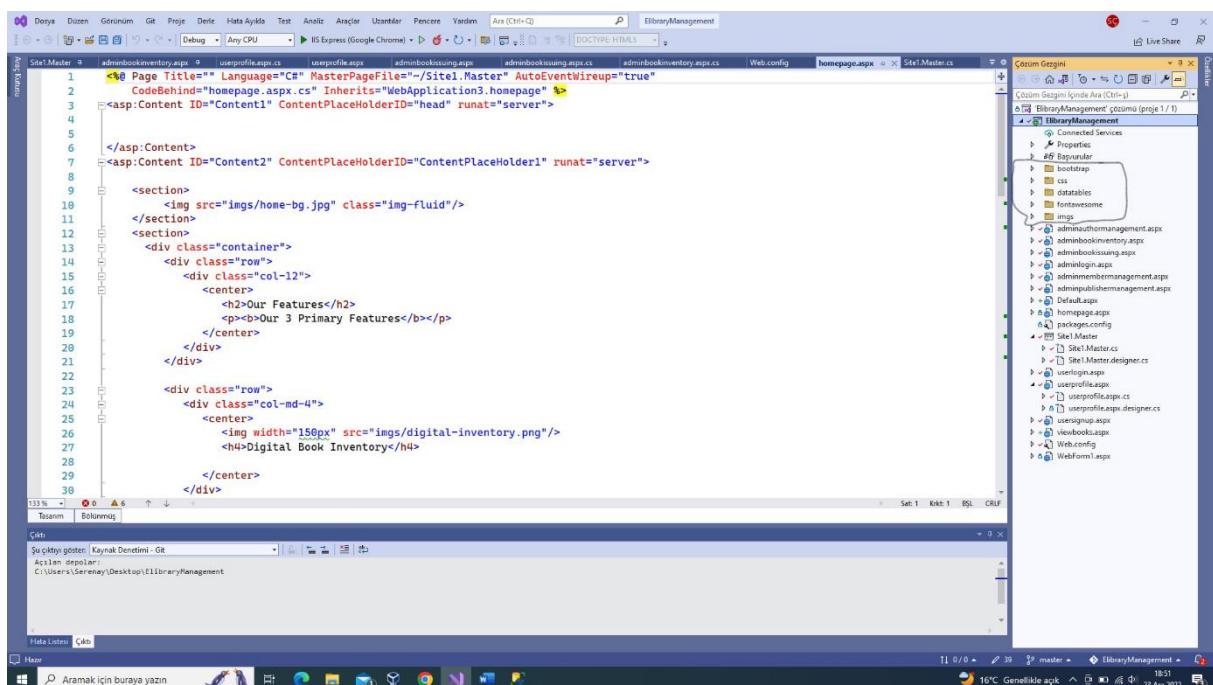
1 – Visual Studio 2022 ve MS SQL Server Management Studio 18 yükleme | Bootstrap ve veriler için veri tabloları kurulumu, ASP.NET’te ana sayfa ve Master Page oluşturma, sayfaya gezinme menüsü ve alt bilgiler ekleme, Bootstrap stili ile ASP.NET’te ana sayfa oluşturma ve sayfaya “imgs” dosyasından (Şekil 1.1’de görülen) görseller ekleme



Şekil 1.1. MS SQL Server Management Studio 18 yüklendi ve veri tabloları eklendi.



Şekil 1.2. ‘imgs’ dosyası oluşturuldu içine kullanmak istediğim görüntüler atıldı.



Şekil 1.3. Bootstrap, datatables, css, font awesome klasörleri oluşturuldu.

Bootstrap, web sayfaları veya uygulamalarında kullanılabilen, HTML ve CSS tabanlı tasarım şablonlarını içerir.

Font Awesome, CSS ve Less'e dayalı bir yazı tipi ve simge araç takımıdır.

Site1.Master'ın içeriği ve kod açıklamaları

```
<%@ Master Language="C#" AutoEventWireup="true"
CodeBehind="Site1.master.cs" Inherits="WebApplication3.Site1" %>

<!DOCTYPE html>

<html>

<head runat="server">

    <title></title>

    <%--bootstrap css--%>

    <link href="bootstrap/css/bootstrap.min.css" rel="stylesheet" />

    <%--datatables css--%>

    <link href="datatables/css/jquery.dataTables.min.css"
rel="stylesheet" />

    <%--fontawesome css--%>

    <link href="fontawesome/css/all.css" rel="stylesheet" />

    <%--our custom css--%>

    <link href="css/customstylesheet.css" rel="stylesheet" />

    <%--jquery--%>

    <script src="bootstrap/js/jquery-3.3.1.slim.min.js"></script>

    <%--popper js--%>

    <script src="bootstrap/js/popper.min.js"></script>

    <%--bootstrap js--%>

    <script src="bootstrap/js/bootstrap.min.js"></script>
```

```
<asp:ContentPlaceHolder ID="head" runat="server">

</asp:ContentPlaceHolder>

</head>

<body>

<form id="form1" runat="server">

<div>

<nav class="navbar navbar-expand-lg navbar-light">

<a class="navbar-brand" href="#">



E-Library

</a>

<button class="navbar-toggler" type="button" data-toggle="collapse"
data-target="#navbarSupportedContent" aria-
controls="navbarSupportedContent" aria-expanded="false" aria-
label="Toggle navigation">

//.navbar-toggler: Daraltma eklentimiz ve diğer gezinme değiştirme
davranışlarımızla kullanım içindir.

// data-toggle: Bir verinin durumunu değiştirmek içindir.

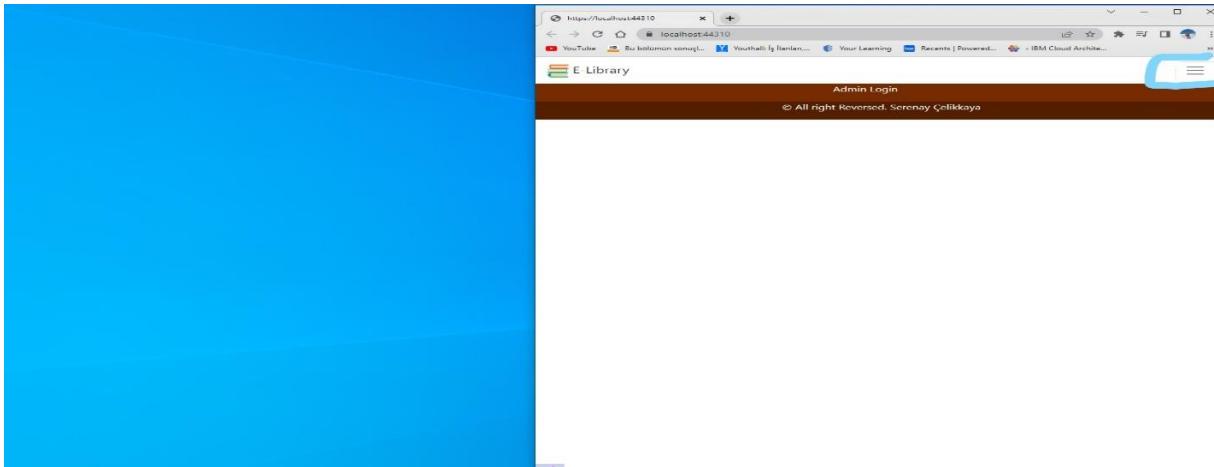
// collapse: navbar içeriğini bir ana kesme noktasına göre
gruplandırmak ve gizlemek içindir.(menü görünümü)

// aria-controls: İçeriği veya varlığı bu özelliğin ayarlandığı öğe
tarafından kontrol edilen öğeyi (veya öğeleri) tanımlar.

// aria-expanded: Bir denetimin genişletilip genişletilmmediğini veya
daraltıldığını ve alt öğelerinin görüntülenip görüntülenmediğini veya
gizlendiğini belirtmek için bir öğede ayarlanır.
```

```
<span class="navbar-toggler-icon"></span>
```

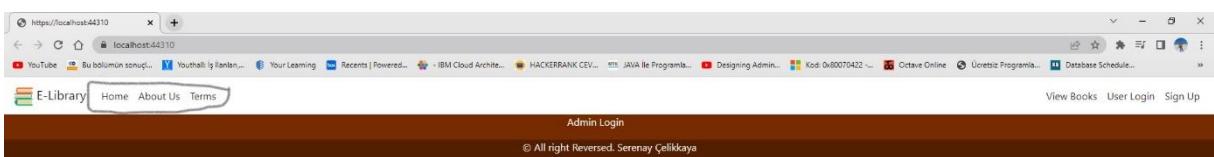
```
</button>
```



Şekil 1.4. Sayfa küçültülse bile görünen kısım değişimeyecek ve menü çubuğu ekleme yapıldı.

```
<div class="collapse navbar-collapse" id="navbarSupportedContent">  
    <ul class="navbar-nav mr-auto" style="list-style-type: none; padding-left: 0;"> //mr-auto, içeriği  
        sağa hizalamayı kullanmamak için sağ kenar boşluğunu otomatik olarak  
        ayarlamak için kullanılır.
```

```
<li class="nav-item active">
```



Şekil 1.5. Sayfaya gezinme linkleri ekleme

```

        <a class="nav-link"
    href="homepage.aspx">Home</a>

    </li>

    <li class="nav-item active">

        <a class="nav-link" href="#">About Us</a>

    </li>

    <li class="nav-item active">

        <a class="nav-link" href="#">Terms</a>

    </li> </ul>

```



Şekil 1.6. Sayfaya gezinme butonları ekleme

```

<ul class="navbar-nav">
<li class="nav-item active">

```

```

        <asp:LinkButton class="nav-link"
ID="LinkButton4" runat="server" OnClick="LinkButton4_Click">View
Books</asp:LinkButton>
    </li>
    <li class="nav-item active">
        <asp:LinkButton class="nav-link"
ID="LinkButton1" runat="server" OnClick="LinkButton1_Click">User
Login</asp:LinkButton>
    </li>
    <li class="nav-item active">
        <asp:LinkButton class="nav-link"
ID="LinkButton2" runat="server" OnClick="LinkButton2_Click">Sign
Up</asp:LinkButton>
    </li>
    <li class="nav-item active">
        <asp:LinkButton class="nav-link"
ID="LinkButton3" runat="server" OnClick="LinkButton3_Click"
Visible="False">Logout</asp:LinkButton> //sayfada görmek istemediğim
butonların visible değerlerini false olarak değiştirdim.
    </li>
    <li class="nav-item active">
        <asp:LinkButton class="nav-link"
ID="LinkButton7" runat="server" OnClick="LinkButton7_Click"
Visible="False">Hello user</asp:LinkButton>
    </li>
</ul>
</div>
</nav>

```

```
</div>
```

```

<!-- Main Content Placeholder -->
<div>
    <asp:ContentPlaceHolder ID="ContentPlaceHolder1"
runat="server">
    </asp:ContentPlaceHolder>

```

```
</div>
```

```

<!-- Main Content Placeholder -->
<!-- Footer --> //sayfaya alt bilgiler eklemek için kullanılır
ama Site1.Master sayfasında sadece Admin Login butonunu görmek
istediğim için diğerlerinin visible değerlerini false olarak
değiştirdim.

<footer>
    <div id="footer1" class="container-fluid">
        <div class="row">
            <div class="col-xs-12 col-sm-12 col-md-12 text-
center">
                <p>
                    <asp:LinkButton class="footerlinks"
ID="LinkButton6" runat="server" OnClick="LinkButton6_Click">Admin
Login</asp:LinkButton>
                    &nbsp;
                    <asp:LinkButton class="footerlinks"
ID="LinkButton11" runat="server" OnClick="LinkButton11_Click"
Visible="False">Author Management</asp:LinkButton>
                    &nbsp;
                    <asp:LinkButton class="footerlinks"
ID="LinkButton12" runat="server" OnClick="LinkButton12_Click"
Visible="False">Publisher Management</asp:LinkButton>
                    &nbsp;
                    <asp:LinkButton class="footerlinks"
ID="LinkButton8" runat="server" OnClick="LinkButton8_Click"
Visible="False">Book Inventory</asp:LinkButton>
                    &nbsp;
                    <asp:LinkButton class="footerlinks"
ID="LinkButton9" runat="server" OnClick="LinkButton9_Click"
Visible="False">Book Issuing</asp:LinkButton>
                    &nbsp;
                    <asp:LinkButton class="footerlinks"
ID="LinkButton10" runat="server" OnClick="LinkButton10_Click"
Visible="False">Member Management</asp:LinkButton>
                </p>
            </div>
        </div>
    </div>
    <div id="footer2" class="container-fluid">

```

```

<div class="row">
    <div class="col-xs-12 col-sm-12 col-md-12 text-
center">
        <p style="color:whiteSmoke">&copy All right
Reversed. <a class="footerlinks" href="#" target="_blank">Serenay
Çelikkaya</a></p>
    </div>
</div>
</div>
</div>
</div>
<!-- ./Footer -->
</form>
</body>

</html>

```



Şekil 1.7. Site1.Master sayfasının görünümü

WebForm1.aspx'in içeriği

```

<%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master"
AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs"
Inherits="WebApplication3.WebForm1" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>

```

```

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
runat="server">
    <h1>
        This is our content webform
    </h1>

</asp:Content>

```

homepage.aspx'in içeriği ve kod açıklamaları

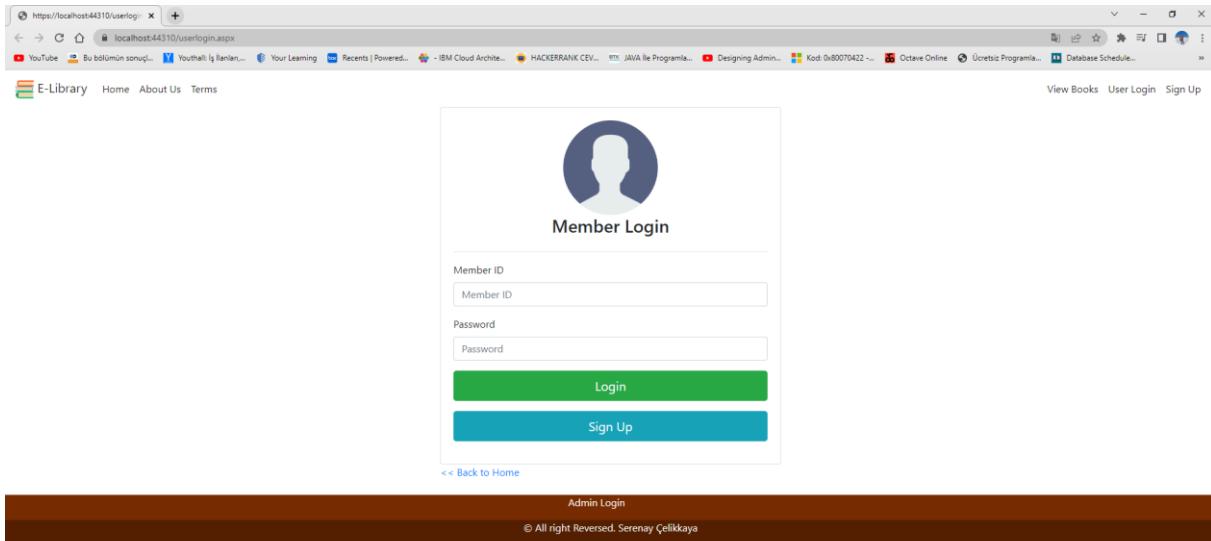
```

<section>
    
</section> // imgs klasöründen resim seçildi. Bir resme .img-fluid
eklediğimizde width:100%; ve height:auto; özelliklerini uygulamış
oluyoruz dolayısıyla resmin genişliği içinde bulunduğu etiketin
genişliğine göre ayarlanmış oluyor.

<div class="row">
    <div class="col-md-4">
        <center>
            
            <h4>Digital Book Inventory</h4> // col-md-4 demek
sayfayı 12 sütunluk bir yer gibi düşünün bunun 4 satırını bir resme
diğer 4 satırını diğer resme ve son 4 satırını da başka bir resme
ayıracaksınız demektir. Böylece sayfanın aynı satırına 3 resim sıradırma
şansınız olacaktır.

2 – ASP.NET'te oturum açma sayfası tasarlama (yönetici ve kullanıcı giriş sayfaları),
Bootstrap Styling ile ASP.NET'te kaydolma/kayıt sayfası oluşturma ve kullanıcı profili
sayfası oluşturma

```



Şekil 2.1. Kullanıcı giriş sayfası ekranı

Kullanıcı veri tabanında kayıtlı ise bilgilerini girerek sisteme giriş yapabilecek ancak kayıtlı değilse önce kaydolması gerekecek.

Şimdi kullanıcı girişü örneği ile veri tabanına bütün bir projenin nasıl bağlandığını göstereceğim.

userlogin.aspx.cs dosyasının içeriği ve kod açıklamaları

```
string strcon =
ConfigurationManager.ConnectionStrings["con"].ConnectionString;
// web.config dosyasıyla bağlantıyı sağlar.
SqlConnection con = new SqlConnection(strcon); //yeni bir sql
bağlantısı ve web.config dosyasındaki connectionStringi kullanarak veri
tabanıyla bağlantıyı sağlar. Yani API burada uygulanmıştır, veri tabanı
ile bağlantı kurmak bunu da frontende yansıtmak API'nin görevidir.
```

web.config dosyasındaki connectionString'in gösterimi

```
<connectionStrings>
<add name="con" connectionString="Data Source=DESKTOP-8UNLIRV;Initial
Catalog=elibraryDB;Integrated Security=true" />
<add name="elibraryDBConnectionString" connectionString="Data
Source=DESKTOP-8UNLIRV;Initial Catalog=elibraryDB;Integrated
Security=True" />
```

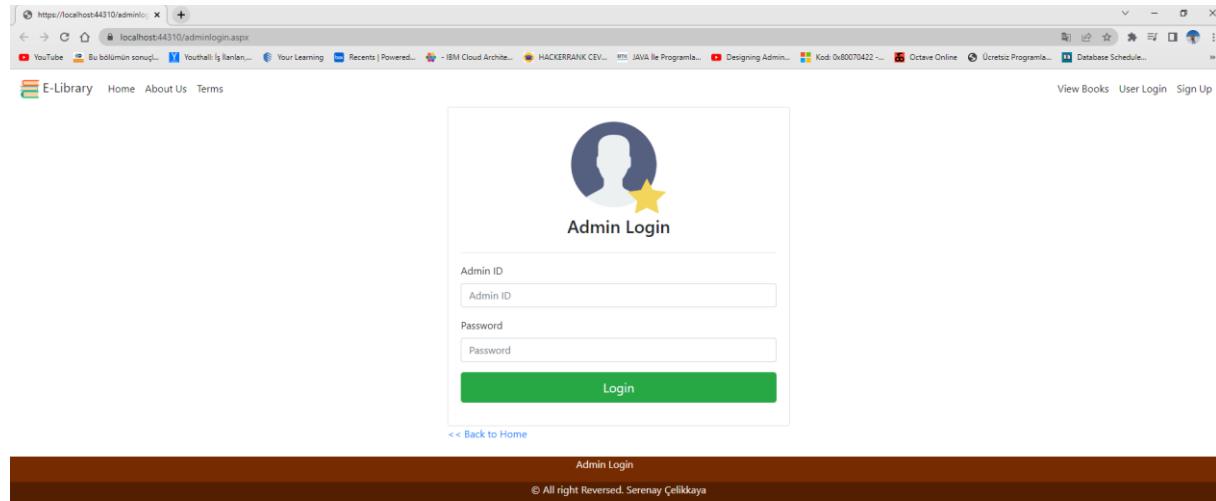
```

    providerName="System.Data.SqlClient" />
<add name="elibraryDBConnectionString2" connectionString="Data
Source=DESKTOP-8UNLIRV;Initial Catalog=elibraryDB;Integrated
Security=True">
    providerName="System.Data.SqlClient" />
</connectionStrings>

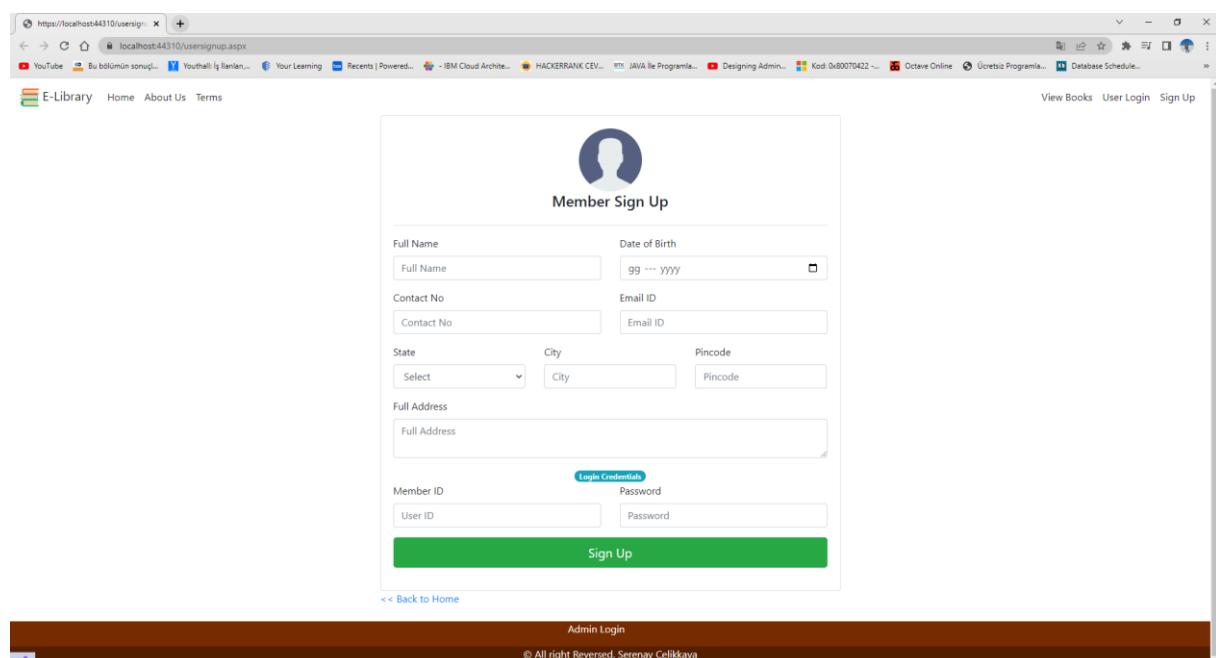
```

Data Source=DESKTOP-8UNLIRV // Bu kısım benim bağlanacağım veri kaynağımın adıdır.

Initial Catalog// Bu kısım veri tabanımın adıdır.



Şekil 2.2. Yönetici (admin) giriş sayfası

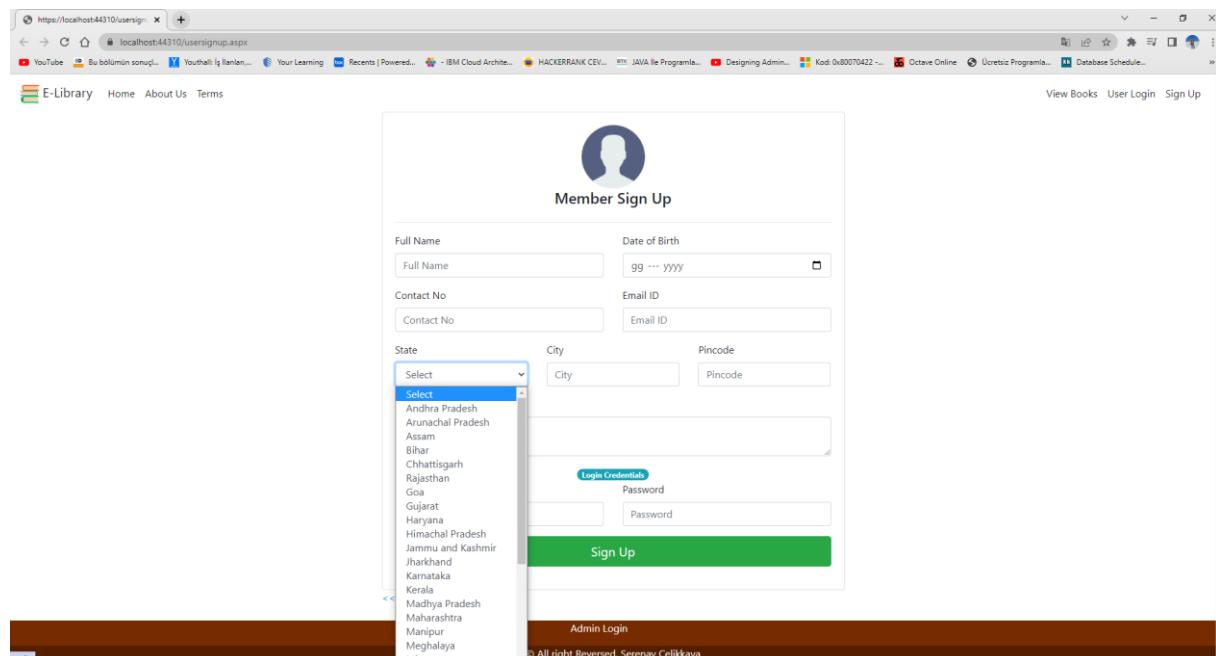


Şekil 2.3. Kayıt olma sayfası

userlogin.aspx dosyasındaki ülke seçimini gösterme

```
<div class="form-group">
    <asp:DropDownList class="form-control"
        ID="DropDownList1" runat="server">
        <asp:ListItem Text="Select" Value="select" />
        <asp:ListItem Text="Andhra Pradesh" Value="Andhra Pradesh" />
        <asp:ListItem Text="Arunachal Pradesh" Value="Arunachal Pradesh" />
        <asp:ListItem Text="Assam" Value="Assam" />
        <asp:ListItem Text="Bihar" Value="Bihar" />
        <asp:ListItem Text="Chhattisgarh" Value="Chhattisgarh" />
        <asp:ListItem Text="Rajasthan" Value="Rajasthan" />
        <asp:ListItem Text="Goa" Value="Goa" />
        <asp:ListItem Text="Gujarat" Value="Gujarat" />
        <asp:ListItem Text="Haryana" Value="Haryana" />
    </asp:DropDownList>
</div>
```

Bu seçimi dropdownlist kullanarak gerçekleştirdim.

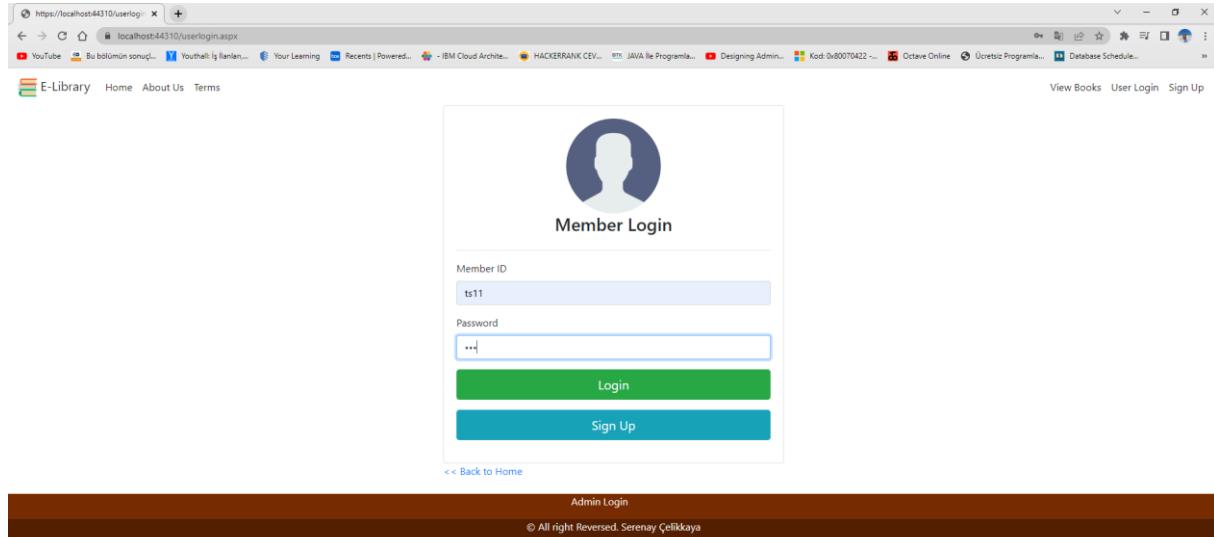


Şekil 2.4. Kayıt olma sayfasındaki ülke seçimi

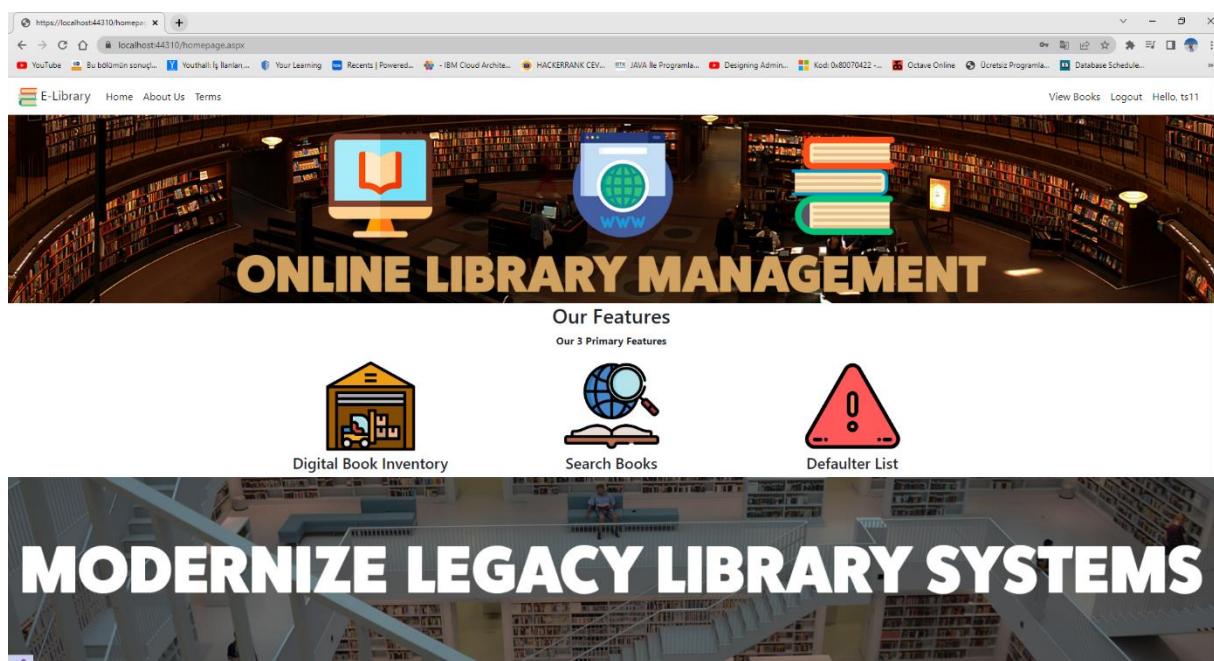
“Login Credentials” kısmının oluşturulmasında badge yani rozetleri kullandım bunlar da bootstrap ögesidir rozetler, görelî yazı tipi boyutlandırması ve birimleri kullanılarak doğrudan üst ögenin boyutuyla eşleşecek şekilde ölçeklenir. badge-pill, rozetleri daha yuvarlak hale getirmek için kullanılır ve badge-info da rozetin yeşil renkli olmasını sağlar.

```
<span class="badge badge-pill badge-info">Login Credentials</span>
```

Userprofile.aspx sayfasını tarayıcıda çalıştırduğumda karşımıza kullanıcının giriş ekranı geliyor, kullanıcı giriş yaptıktan sonra kendi okuduğu kitaplara erişebilme yetisi kazanıyor. Kayıtlı bir kullanıcıyla giriş yapmak için veri tabanındaki dbo.member_master_table isimli tablodan herhangi bir kullanıcının id'si ve şifresi ile giriş yapılabilir.



Şekil 2.5. Burada id'si “ts11” şifresi “123” olan kullanıcı sisteme giriş yapıyor.



Şekil 2.6. Kullanıcı karşısına ana sayfa geliyor, buradan “Hello, ts11” butonuna tıklayarak kendi profiline ulaşabiliyor.

The screenshot shows the 'Your Profile' section with the following details:

- Full Name:** Tanmay Sakpal
- Date of Birth:** 18 Eki 1993
- Contact No:** 5665656565
- Email ID:** adad@exy.com
- State:** Maharashtra
- City:** Mumbai
- Pincode:** 333333
- Full Address:** Burdur
- User ID:** ts11
- Old Password:** 123
- New Password:** New Password

A blue 'Update' button is at the bottom.

The right side shows the 'Your Issued Books' section with one entry:

member_id	member_name	book_id	book_name	issue_date	due_date
ts11	Tanmay Sakpal	b001	The Power of Your Subconscious Mind	2022-12-15	2022-12-20

Showing 1 to 1 of 1 entries

Şekil 2.7. Kullanıcı profili sayfası

3 - Bootstrap Styling ile ASP.NET'te yazar ve yayıncı yönetimi sayfaları tasarlama, yöneticinin kitabı vermesi için sayfa tasarlama, yöneticinin (adminin) üyeleri ve kitabı envanterlerini görmesi için sayfa tasarlama

The screenshot shows the 'Author Details' section with the following data:

Author ID	a001	Author Name	Joseph Murphy
-----------	------	-------------	---------------

Buttons: Add (green), Go (blue), Update (yellow), Delete (red).

The right side shows the 'Author List' section with two entries:

author_id	author_name
a002	Dale Carnegie
a003	Napolean Hill

Showing 1 to 2 of 2 entries

Şekil 3.1. Veri tabanına “a001” id numaralı “Joseph Morphy” adında yazar ekliyorum.



Şekil 3.2. Ekranda “Author added successfully” mesajını görüyoruz.

A screenshot of a web application interface. On the left, there's a 'Author Details' form with fields for 'Author ID' (ID: a001) and 'Author Name' (Name: Joseph Murphy). It has buttons for 'Add', 'Update', and 'Delete'. On the right, there's a 'Author List' table showing three entries: a001 (Joseph Murphy), a002 (Dale Carnegie), and a003 (Napoleon Hill). The table includes a dropdown for 'Show' (set to 10) and a search bar. At the bottom, there's a footer with links for 'View Books', 'User Login', and 'Sign Up', along with copyright information: '© All right Reversed. Serenay Çelikkaya'.

Şekil 3.3. Eklenen yazarın sayfada görünen hali

Burada gördüğünüz sayfaların C# kodları aşağıda verilmiştir. Gördüğünüz kod, veri tabanına yeni bir yazar eklerken, yazar bilgilerini güncellerken, silerken ve bir yazarı görüntüleme ve bulurken yapılan işlemlerdir. Yeni bir yazarı veri tabanına ismi ve id numarası ile ekleyebiliyoruz ve ekleme başarılı ise “Author added successfully” mesajını alıyoruz. Bazı kod satırlarının açıklamaları yanlarında yorum olarak belirtilmiştir.

```

void addNewAuthor()
{
    try
    {
        SqlConnection con = new SqlConnection(strcon); // sql'e
bağlantı kuruldu.
        if (con.State == ConnectionState.Closed)
        {
            con.Open();
        }

        SqlCommand cmd = new SqlCommand("INSERT INTO
author_master_tbl" +
                "(author_id,author_name)
values(@author_id,@author_name)", con); // insert into methodu ile yeni
bir yazar veri tabanına eklendi.

        cmd.Parameters.AddWithValue("@author_id", TextBox1.Text.Trim());
        cmd.Parameters.AddWithValue("@author_name",
TextBox2.Text.Trim());

        cmd.ExecuteNonQuery();
        con.Close();
        Response.Write("<script>alert('Author added
Successfully');</script>");
        clearForm();
        GridView1.DataBind(); // elde edilen DataSet'i GridView
kontrolüne bağladım..
    }
    catch (Exception ex)
    {
        Response.Write("<script>alert('" + ex.Message +
"');</script>");
    }
}

```

Eğer id numarası aynı olan başka bir yazar eklemeye çalışırsak onun kontrolünü aşağıdaki koşul ile yapıp şu uyarıyı alacağız:

```

if (checkIfAuthorExists())
{
    Response.Write("<script>alert('Author with this ID
already Exist. You cannot add " +
                "another Author with the same Author
ID');</script>");
}
else
{
    addNewAuthor();
}

```

Şimdi bir yazarı veri tabanından silme işlemine bakalım.

The screenshot shows a web application interface. On the left, the 'Author Details' page has a green circular icon with a person silhouette. Below it, 'Author ID' is set to 'a001' and 'Author Name' is 'Joseph Murphy'. There are 'Add', 'Update', and 'Delete' buttons. On the right, the 'Author List' page shows a table with three entries: a001 (Joseph Murphy), a002 (Dale Carnegie), and a003 (Napolean Hill). A search bar and pagination buttons ('Previous', '1', 'Next') are also present.

Şekil 3.4. "a001" id numarasını girip "Go" butonuna basılacak.

This screenshot is identical to the one above, showing the 'Author Details' page with 'a001' and 'Joseph Murphy' and the 'Author List' page with the same three entries. This indicates that the action of clicking the 'Go' button has not yet been performed.

Şekil 3.5. "Go" butonuna bastıktan sonra yazarın adı(Joseph Morphy) karşımıza geliyor.



Şekil 3.6. “Delete” butonuna basınca ekranda “Authot deleted succesfully” mesajını görüyoruz.

A screenshot of a web application interface. On the left, there is a 'Author Details' form with fields for 'Author ID' (set to 'a001') and 'Author Name' (empty), and buttons for 'Add', 'Update', and 'Delete'. On the right, there is an 'Author List' table showing two entries: 'a002 Dale Carnegie' and 'a003 Napoleon Hill'. The table includes a dropdown for 'Show [10] entries', a search bar, and navigation buttons for 'Previous' and 'Next'. At the bottom, there is a brown footer bar with the text 'Admin Login' and '© All right Reversed. Serenay Çelikkaya'.

Şekil 3.7. Sildiğimiz yazarı ona ait id numarası ile sorgulamak isteyince ekranda göreceğimiz mesaj için gösterilmiştir.



Şekil 3.8. “Invalid Author ID” mesajını alıyoruz.

Aşağıda verilen kod silme işlemine aittir.

```
void deleteAuthor()
{
    try
    {
        SqlConnection con = new SqlConnection(strcon);
        if (con.State == ConnectionState.Closed)
        {
            con.Open();
        }

        SqlCommand cmd = new SqlCommand("DELETE from
author_master_tbl WHERE author_id='" + TextBox1.Text.Trim() + "'",
con);

        cmd.ExecuteNonQuery();
        con.Close();
        Response.Write("<script>alert('Author Deleted
Successfully');</script>");
        clearForm();
        GridView1.DataBind();

    }
    catch (Exception ex)
    {
        Response.Write("<script>alert('" + ex.Message +
"');</script>");
    }
}
```

Bu kodda da sayfada gördüğümüz aradığımız id numarasına ait biri yokken aldığımız mesaj gösterilmiştir.

```
void getAuthorByID()
{
```

```

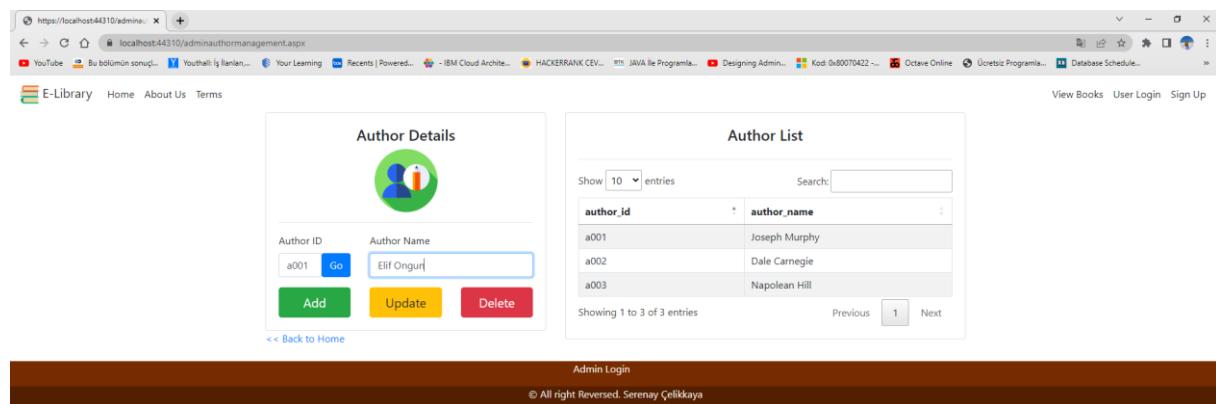
try
{
    SqlConnection con = new SqlConnection(strcon);
    if (con.State == ConnectionState.Closed)
    {
        con.Open();
    }

    SqlCommand cmd = new SqlCommand("SELECT * from
author_master_tbl where author_id='" + TextBox1.Text.Trim() + "' ;",
con);
    SqlDataAdapter da = new SqlDataAdapter(cmd);
    DataTable dt = new DataTable();
    da.Fill(dt);

    if (dt.Rows.Count >= 1)
    {
        TextBox2.Text = dt.Rows[0][1].ToString();
    }
    else
    {
        Response.Write("<script>alert('Invalid Author
ID');</script>");
    }
}

catch (Exception ex)
{
    Response.Write("<script>alert('" + ex.Message +
"');</script>");
}
}

```



Şekil 3.9. “a001” id numaralı yazarın adını “Elif Ongun” olarak güncelleme

Kodda güncelleme için eğer yazar bulunuyorsa güncelleyeceği bulunmuyorsa uyarı mesajı alacağımı gösterdim.

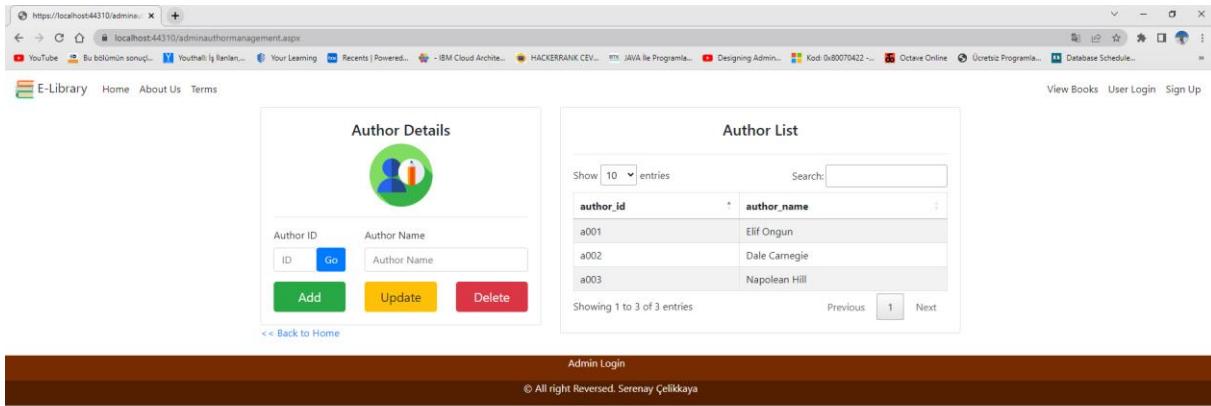
Update button için:

```
protected void Button3_Click(object sender, EventArgs e)
{
    if (checkIfAuthorExists())
    {
        updateAuthor();

    }
    else
    {
        Response.Write("<script>alert('Author does not exist');</script>");
    }
}
```



Şekil 3.10. Güncelleme işlemi başarılı mesajını aldım.



Şekil 3.11. Güncelleme yapıldıktan sonra yazarın güncel durumu gösterildi.

Kodda id numarasına göre güncelleme işlemi yapılmıştır.

```
void updateAuthor()
{
    try
    {
        SqlConnection con = new SqlConnection(strcon);
        if (con.State == ConnectionState.Closed)
        {
            con.Open();
        }

        SqlCommand cmd = new SqlCommand("UPDATE
author_master_tbl SET author_name=@author_name WHERE author_id='"
+ TextBox1.Text.Trim() + "'", con);

        cmd.Parameters.AddWithValue("@author_name",
TextBox2.Text.Trim());

        cmd.ExecuteNonQuery();
        con.Close();
        Response.Write("<script>alert('Author Updated
Successfully');</script>");
        clearForm();
        GridView1.DataBind();
    }
    catch (Exception ex)
    {
        Response.Write("<script>alert('" + ex.Message +
"');</script>");
    }
}
```

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, the database 'elibraryDB' is selected. In the center pane, a query window displays the following script and its results:

```

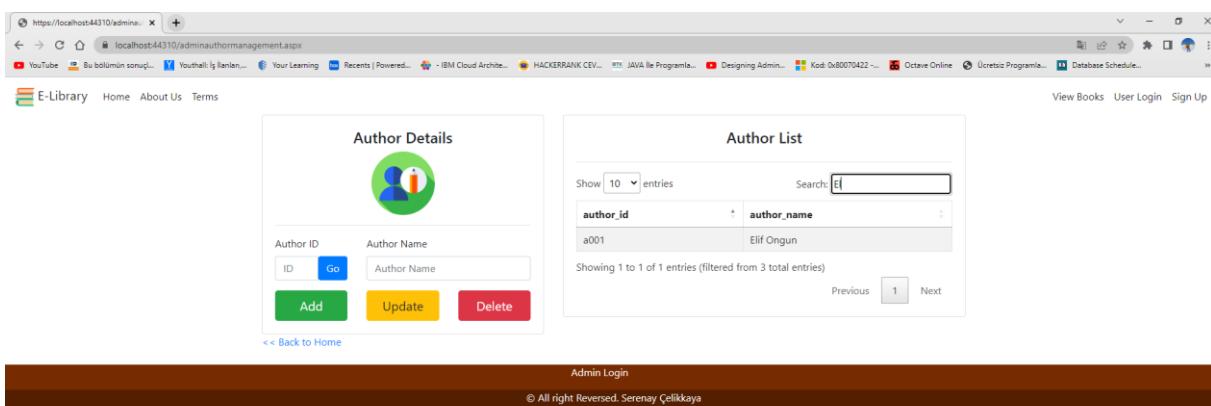
SELECT TOP (1000) [author_id]
      ,[author_name]
   FROM [elibraryDB].[dbo].[author_master_tbl]
  
```

The results grid shows four rows of data:

author_id	author_name
a001	Elif Ongun
a002	Dale Carnegie
a003	Napoleon Hill
a004	Mark Manson

At the bottom of the screen, a message indicates: "Query executed successfully."

Şekil 3.12. Veri tabanında görünen dbo.author_master_tbl tablosunun son hali



Şekil 3.13. Sayfada “Search” kısmına yazarın adını yazarak yazarı aratabiliyoruz.

Bu kod yazarın varlığını görmek yanı yazarı arama işlemi için yazılmıştır.

```

bool checkIfAuthorExists()
{
    try
    {
        SqlConnection con = new SqlConnection(strcon);
        if (con.State == ConnectionState.Closed)
        {
            con.Open();
        }
    }
}

```

```

SqlCommand cmd = new SqlCommand("SELECT * from
author_master_tbl where author_id='"
+ TextBox1.Text.Trim() + "';", con);
SqlDataAdapter da = new SqlDataAdapter(cmd);
DataTable dt = new DataTable();
da.Fill(dt);

if (dt.Rows.Count >= 1)
{
    return true;
}
else
{
    return false;
}

}

catch (Exception ex)
{
    Response.Write("<script>alert('" + ex.Message +
"');</script>");
    return false;
}
}

```

Buraya kadar gördüğümüz kodlar yayıncı için de yazılmış olup sayfa görünümü şu şekildedir:

The screenshot shows a web application interface. On the left, there is a 'Publisher Details' form with a circular icon containing a book symbol. It has fields for 'Publisher ID' (p001) and 'Publisher Name' (Amazing Reads), with 'Go' and 'Add' buttons. On the right, there is a 'Publisher List' table with two entries: p001 (Amazing Reads) and p002 (Harper Collins). The table includes a search bar and navigation buttons for 'Previous' and 'Next'. At the bottom, there is a dark footer bar with 'Admin Login' and '© All right Reversed. Serenay Çelikkaya' text.

publisher_id	publisher_name
p001	Amazing Reads
p002	Harper Collins

Şekil 3.14. Yayıncı sayfası

The screenshot shows the Object Explorer on the left with the 'libraryDB' database selected. In the center, a query window displays the following SQL script:

```
***** Script for SelectTopNRows command from SSMS *****
SELECT [publisher_id]
      ,[publisher_name]
   FROM [libraryDB].[dbo].[publisher_master_table]
```

The results pane shows two rows of data:

publisher_id	publisher_name
a001	Hawking Reads
a002	Hawker Collins

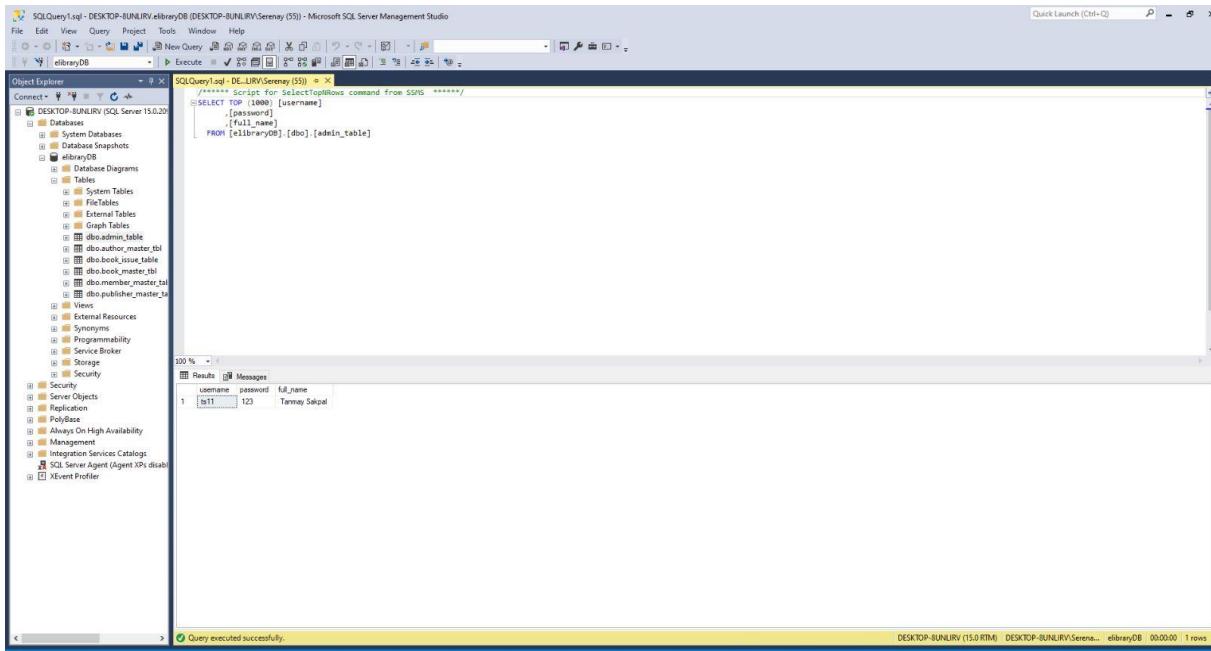
Şekil 3.15. Yayıncı bilgilerinin kaydedildiği dbo.publisher_master_table tablosu

Şimdi yöneticinin kitabı vermesi için yöneticinin sisteme nasıl giriş yaptığına bakalım.

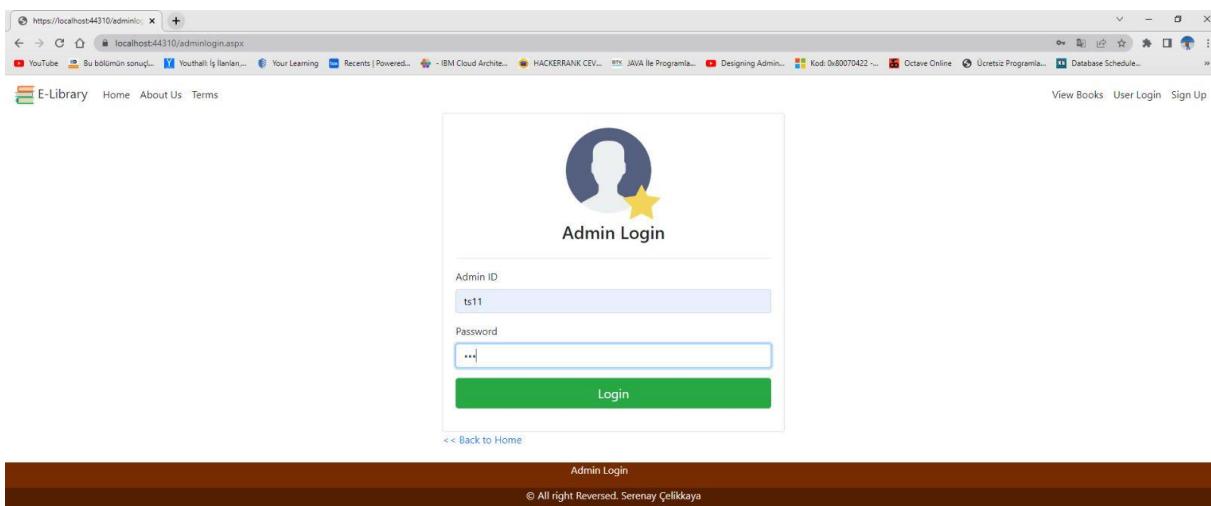


Şekil 3.16. Anasayfada görülen “Admin Login” butonuna tıklıyoruz.

Yöneticinin sisteme girişi, açılan ekranda veri tabanına kaydettiğimiz bilgileri girerek sağlanıyor.



Şekil 3.17. Veri tabanında dbo.admin_table'daki yönetici bilgileri



Şekil 3.18. Bilgiler girilerek yönetici sisteme giriş yapıyor.

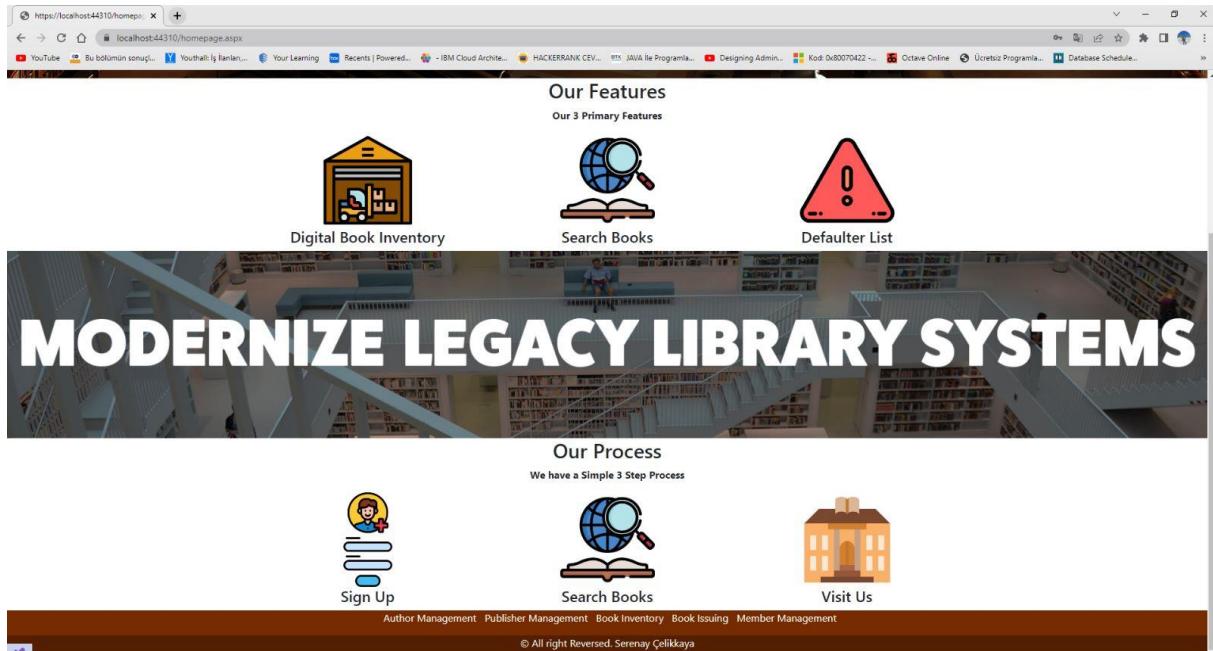
```
// login button click
event
```

```
protected void
Button1_Click(object
sender, EventArgs e)
{
try
{
SqlConnection con = new SqlConnection(strcon);
```

```

if (con.State == ConnectionState.Closed)
{
    con.Open();
}
SqlCommand cmd = new SqlCommand("select * from
admin_table where username='"
+ TextBox1.Text.Trim() + "' AND password='"
+ TextBox2.Text.Trim() + "'", con);
SqlDataReader dr = cmd.ExecuteReader();
if (dr.HasRows)
{
    while (dr.Read())
    {
        Response.Write("<script>alert('Successful
login');</script>");
        Session["username"] =
            dr.GetValue(0).ToString(); //dbo.admin_table
        tablosunun 1. Sütununu gösterir.
        Session["fullname"] =
            dr.GetValue(2).ToString();
        Session["role"] = "admin"; // tablonun
        3.sütununu gösterir.
    }
    Response.Redirect("homepage.aspx"); // giriş
    yapıldıktan sonra anasayfaya yönlendirilir.
}
else
{
    Response.Write("<script>alert('Invalid
credentials');</script>");
}
}
catch (Exception ex)
{
    Response.Write("<script>alert('" + ex.Message +
"');</script>");
}
}

```



Şekil 3.19. Yöneticiye sisteme giriş yaptıktan sonra anasayfada yazar, yayıncı bilgilerini güncelleyebilmesi için, yeni kitaplar eklemesi ve kitapların kullanıcıya verilmesi için ve son olarak da kullanıcı bilgilerini güncellemesi için sadece yöneticinin sahip olduğu yetkinlikler verilmiştir.

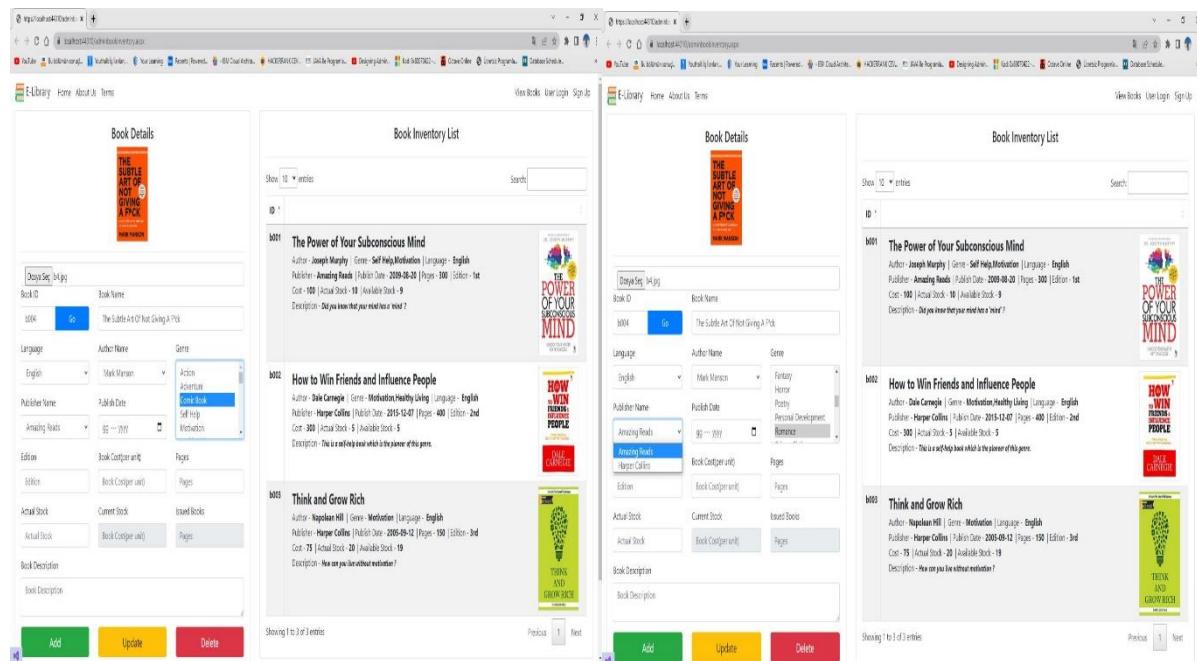
Yönetici kişinin kitap bilgilerini sisteme kaydetmesini inceleyelim.

Şekil 3.20. Bu sayfada kitap bilgilerini girerken kitabın kapak resmini koymak için “Dosya Seç” butonu ile “imgs” klasöründen kitap resmi seçiyorum.

Kitabın resmini seçme işlemi bu kod sayesinde yapıldı.

```
string filename = Path.GetFileName(FileUpload1.PostedFile.FileName);
FileUpload1.SaveAs(Server.MapPath("imgs/" + filename));
filepath = "~/imgs/" + filename;
```

```
cmd.Parameters.AddWithValue("@book_img_link", filepath); // bu komut
ile de görsel linki veri tabanına kaydedilmiş oldu.
```



Şekil 3.21. Kitabın diğer bilgilerini ekleme ekranı

Resimlerde gördüğümüz üzere kitabı türünü seçerken çoklu seçim yapabiliyoruz. Bunu da şu şekilde sağladım:

```
String
genres="";
foreach (int i in ListBox1.GetSelectedIndices())
{
    genres = genres + ListBox1.Items[i] + ",";
}
// genres = Adventure,Self Help,
genres = genres.Remove(genres.Length - 1);
```

Yaynıcı ve yazarın bilgilerini veri tabanından alıyorum.

```
SqlCommand cmd = new
SqlCommand("SELECT
author_name from
author_master_tbl;",
con);
SqlDataAdapter da = new SqlDataAdapter(cmd);
DataTable dt = new DataTable();
da.Fill(dt);
DropDownList3.DataSource = dt;
DropDownList3.DataValueField = "author_name";
DropDownList3.DataBind();
```

```

cmd = new SqlCommand("SELECT publisher_name
from publisher_master_table;", con);
da = new SqlDataAdapter(cmd);
dt = new DataTable();
da.Fill(dt);
DropDownList2.DataSource = dt;
DropDownList2.DataValueField =
"publisher_name";
DropDownList2.DataBind();

```

Stok sayısını yazarken “Convert” fonksiyonunu kullanıyorum çünkü veri tabanına bu bilgileri varchar türünde kaydetmiştim.

```

global_actual_stock =
Convert.ToInt32(dt.Rows[0]["actua
l_stock"].ToString().Trim());
global_current_stock =
Convert.ToInt32(dt.Rows[0]["curre
nt_stock"].ToString().Trim());

```



The screenshot shows a browser window with the URL 'https://localhost:44310/admin...'. The page title is 'localhost:44310/adminbookinventory.aspx'. The main content area displays the message 'localhost:44310 web sitesinin mesajı' and 'Book added successfully.' Below the message is a blue 'Tamam' button.

Şekil 3.22. Kitap bilgileri sisteme başarıyla eklendi.

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, the database 'elibraryDB' is selected. In the center pane, a query window displays the following T-SQL code:

```

SELECT TOP 1000 [book_id]
      ,[book_name]
      ,[genre]
      ,[author_name]
      ,[publisher_name]
      ,[publisht_date]
      ,[language]
      ,[edition]
      ,[book_cost]
      ,[no_of_pages]
      ,[book_description]
      ,[actual_stock]
      ,[current_stock]
      ,[book_img_link]
  FROM [elibraryDB].[dbo].[book_master_tbl]

```

The results grid below shows four rows of book data:

book_id	book_name	genre	author_name	publisher_name	publisht_date	language	edition	book_cost	no_of_pages	book_description	actual_stock	current_stock	book_img_link
b001	The Power of Your Subconscious Mind	Self Help,Motivation	Joseph Murphy	Amazing Reads	2009-08-20	English	1st	100	300	Did you know that your mind has a 'mind'? This is a self-help book which is the pioneer o...	10	9	~\img\b1.jpg
b002	How to Win Friends and Influence People	Motivation,Healthy Living	Dale Carnegie	Harper Collins	2015-12-07	English	2nd	300	400	This is a self-help book which is the pioneer o...	5	5	~\img\b2.jpg
b003	Think and Grow Rich	Motivation	Napoleon Hill	Harper Collins	2005-09-12	English	3rd	75	150	How can you live without motivation?	20	19	~\img\b3.jpg
b004	The Subtle Art Of Not Giving A F*ck	Comic Book,Romance	Mark Manson	Harper Collins	2020-02-12	English	1st	100	300	Nice and fun	50	49	~\img\b4.jpg

At the bottom of the results grid, it says "Query executed successfully." and shows connection details: DESKTOP-BUNLURV (15.0 RTM) | DESKTOP-BUNLURV.Serenay | elibraryDB | 00:00:00 | 4 rows.

Şekil 3.23. Veri tabanına yeni eklenen kitap kaydedildi.

The screenshot shows a web-based application for managing a book inventory. On the left, there is a 'BOOK DETAILS' form with fields for Book ID (b004), Book Name (The Subtle Art Of Not Giving A F*ck), Language (English), Author Name (Mark Manson), Publisher Name (Harper Collins), Publish Date (12 Şubat 2020), Edition (1st), Book Cost(per unit) (100), Pages (300), Actual Stock (50), Current Stock (50), and Book Description (Nice and fun). There are buttons for 'Add', 'Update', and 'Delete'.

On the right, there is a 'BOOK INVENTORY LIST' table with the following data:

ID	Book Name	Author Name	Publisher Name	Publish Date	Language	Edition	Book Cost	Pages	Description	Actual Stock	Current Stock	Image
b001	The Power of Your Subconscious Mind	Joseph Murphy	Amazing Reads	2009-08-20	English	1st	100	300	Did you know that your mind has a 'mind'? This is a self-help book which is the pioneer o...	10	9	~\img\b1.jpg
b002	How to Win Friends and Influence People	Dale Carnegie	Harper Collins	2015-12-07	English	2nd	300	400	This is a self-help book which is the pioneer o...	5	5	~\img\b2.jpg
b003	Think and Grow Rich	Napoleon Hill	Harper Collins	2005-09-12	English	3rd	75	150	How can you live without motivation?	20	19	~\img\b3.jpg
b004	The Subtle Art Of Not Giving A F*ck	Mark Manson	Harper Collins	2020-02-12	English	1st	100	300	Nice and fun	50	49	~\img\b4.jpg

Şekil 3.24. Sayfada yeni eklenen kitabı görüyoruz.

Yöneticinin kitabı vermesi için tasarlanan sayfanın işlevine bakalım.

The screenshot shows a web application interface for a library system. On the left, the 'Book Issuing' page has fields for Member ID (sero17), Book ID (b004), and Issue Date (99-12-2022). A 'Go' button is present. On the right, the 'Issued Book List' page displays a table with two entries:

member_id	member_name	book_id	book_name	issue_date	due_date
ts11	Tanmay Sakpal	b001	The Power of Your Subconscious Mind	2022-12-15	2022-12-20
zeynep	Zeynep Yıldız	b003	Think and Grow Rich	2022-12-12	2022-12-20

Below the table, it says 'Showing 1 to 2 of 2 entries'. At the bottom, there are links for Author Management, Publisher Management, Book Inventory, Book Issuing, Member Management, and a copyright notice: '© All right Reversed. Serenay Çelikkaya'.

Şekil 3.25. Member ID ve Book ID kısımları yazılıp “Go” butonuna basıldıktan sonra karşımıza kullanıcının ve kitabı ismi geliyor.

Aşağıda görülen kodda nasıl yapıldığı açıklanmıştır.

```

SqlCommand cmd = new
SqlCommand("select book_name
from book_master_tbl WHERE
book_id=''" +
TextBox1.Text.Trim() + "'", con);
SqlDataAdapter da = new
SqlDataAdapter(cmd);
DataTable dt = new DataTable();
da.Fill(dt);
if (dt.Rows.Count >= 1)
{
    TextBox4.Text =
    dt.Rows[0]["book_name"].ToString();
}
else
{
    Response.Write("<script>alert('Wrong
Book ID');</script>");
}
cmd = new SqlCommand("select
full_name from member_master_table
WHERE member_id=''" +
TextBox2.Text.Trim() + "'", con);
da = new SqlDataAdapter(cmd);
dt = new DataTable();
da.Fill(dt);

```

```

if (dt.Rows.Count >= 1)
{
    TextBox3.Text =
        dt.Rows[0]["full_name"].ToString();
}
else
{
    Response.Write("<script>alert('Wrong
User ID');</script>");
}

```

The screenshot shows a web application interface. On the left, there is a 'Book Issuing' form with fields for Member ID (sero17), Book ID (b004), Member Name (Serenay Çelikkaya), Book Name (The Subtle Art Of Not Giving A F*ck), Issue Date (15 Mar 2022), and Due Date (25 Mar 2022). Below the form are two buttons: 'Issue' (blue) and 'Return' (green). On the right, there is a table titled 'Issued Book List' displaying three entries:

member_id	member_name	book_id	book_name	issue_date	due_date
sero17	Serenay Çelikkaya	b004	The Subtle Art Of Not Giving A F*ck	2022-03-15	2022-03-25
ts11	Tanmay Sakpal	b001	The Power of Your Subconscious Mind	2022-12-15	2022-12-20
zeynepy	Zeynep Yıldız	b003	Think and Grow Rich	2022-12-12	2022-12-20

At the bottom of the page, there is a navigation bar with links: Author Management, Publisher Management, Book Inventory, Book Issuing, Member Management, and a copyright notice: © All right Reserved. Serenay Çelikkaya.

Şekil 3.26. Kitabın verileceği tarih aralıkları girildikten sonra sayfada verilen kitapların listesini görebiliyoruz.

Kitap vermenin kodu aşağıda gösterilmiştir.

```

SqlCommand cmd = new SqlCommand("INSERT INTO
book_issue_table(member_id,member_name,book_id,book_name,issue_date,due
_date)
values(@member_id,@member_name,@book_id,@book_name,@issue_date,@due_dat
e)", con);

cmd.Parameters.AddWithValue("@member_id",
TextBox2.Text.Trim());
cmd.Parameters.AddWithValue("@member_name",
TextBox3.Text.Trim());
cmd.Parameters.AddWithValue("@book_id",
TextBox1.Text.Trim());

```

```

        cmd.Parameters.AddWithValue("@book_name",
TextBox4.Text.Trim());
        cmd.Parameters.AddWithValue("@issue_date",
TextBox5.Text.Trim());
        cmd.Parameters.AddWithValue("@due_date",
TextBox6.Text.Trim());

        cmd.ExecuteNonQuery();

        cmd = new SqlCommand("update book_master_tbl set
current_stock = current_stock-1 WHERE book_id='" + TextBox1.Text.Trim()
+ "'", con);

        cmd.ExecuteNonQuery();

        con.Close();
        Response.Write("<script>alert('Book Issued
Successfully');</script>");
    }
}

```



Şekil 3.27. Sayfada görülen mesaj

```

        GridView1.DataBind();
    }
}

```

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, the database 'elibraryDB' is selected. In the center pane, a query window displays the following T-SQL script:

```

SELECT TOP (1000) [member_id]
      ,[member_name]
      ,[book_id]
      ,[book_name]
      ,[issue_date]
      ,[due_date]
  FROM [elibraryDB].[dbo].[book_issue_table]

```

The results pane shows the following data:

member_id	member_name	book_id	book_name	issue_date	due_date
ts11	Tanay Sakal	b001	The Power of Your Subconscious Mind	2022-12-15	2022-12-20
zeynep	Zeynep Yıldız	b003	Think and Grow Rich	2022-12-12	2022-12-20
sero17	Sereny Celikkaya	b004	The Subtle Art Of Not Giving A Fck	2022-03-15	2022-03-25

At the bottom of the results pane, it says "Query executed successfully."

Şekil 3.28. Verilen kitapların kaydedildiği dbo.book_issue_table tablosu

4 – Kayıt sayfası oluşturma, kullanıcı giriş sayfası, üye verilerini yönetici kullanıcının güncellemesi/silebilmesi

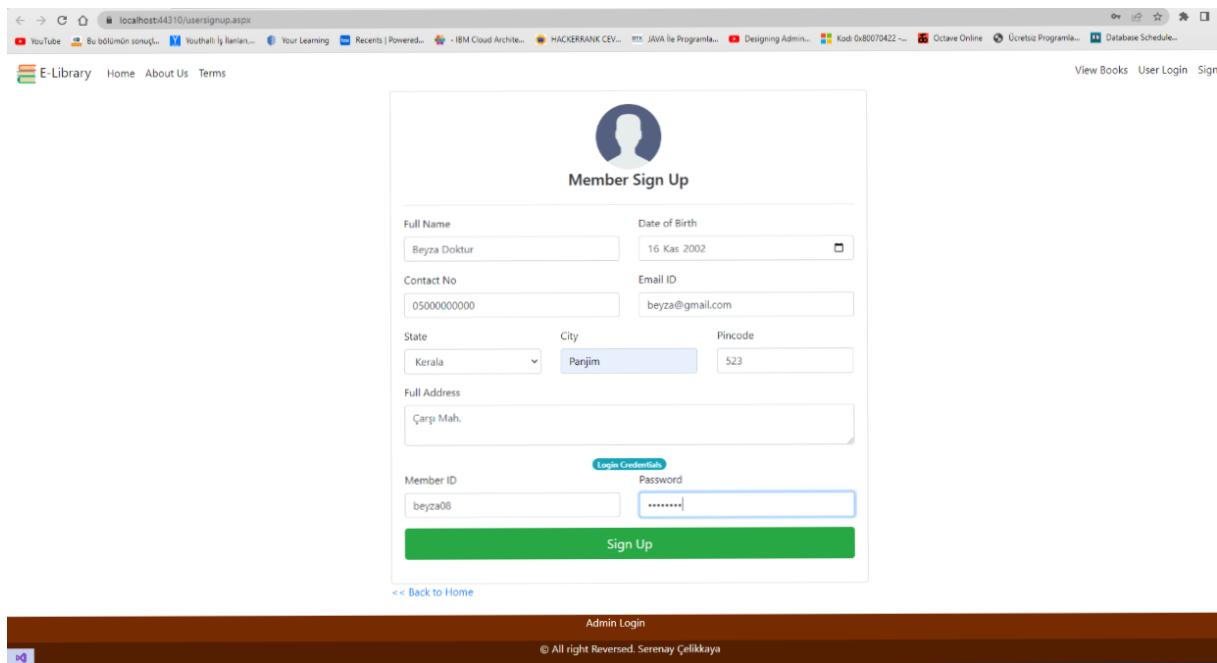
Kayıt sayfası ve kullanıcı kaydetme işlemlerini inceleyelim.

The screenshot shows a web browser window with the URL <https://localhost:44310/usersignup>. The page title is "Member Sign Up". The form fields include:

- Full Name: Full Name (text input), Date of Birth (date input)
- Contact No: Contact No (text input), Email ID (text input)
- State: Select (dropdown), City: City (text input), Pincode: Pincode (text input)
- Full Address: Full Address (text input)
- Login Credentials: Member ID (User ID text input), Password (Password text input)
- Sign Up button (green button)

Below the form, there is a link "<< Back to Home" and a dark brown footer bar with the text "Admin Login".

Şekil 4.1. Kullanıcı kaydetme sayfası



Şekil 4.2. Yeni kullanıcı kaydetme

Kullanıcının kayıtlı olup olmadığını kontrol ediyoruz.

```

Bool
checkMemberExists()
{
    try
    {
        SqlConnection con = new SqlConnection(strcon);
        if (con.State == ConnectionState.Closed)
        {
            con.Open();
        }
        SqlCommand cmd = new SqlCommand("SELECT * from
        member_master_table where member_id='"
        TextBox8.Text.Trim() + "';", con);
        SqlDataAdapter da = new SqlDataAdapter(cmd);
        DataTable dt = new DataTable();
        da.Fill(dt);
        if (dt.Rows.Count >= 1) // 9. sütunda kayıtlı
        veri var mı kontrolü
        {
            return true;
        }
        else
        {
            return false;
        }
    }
    catch (Exception ex)
}

```

```

    {
        Response.Write("<script>alert('" + ex.Message +
        "');</script>");
        return false;
    }
}

```

Eğer kullanıcı önceden kayıt olmuşsa;

```

protected void
Button1_Click(object
sender, EventArgs e)
{
    if (checkMemberExists())
    {
        Response.Write("<script>alert('Member Already
Exist with this Member ID, try other
ID');// bu uyarıyı alacağız
    }
    else
    {
        signUpNewMember();
    }
}

```

Yeni kullanıcıyı kayıt etmek için:

```

SqlConnect
ion con =
new
SqlConnect
ion(strcon
);
if (con.State == ConnectionState.Closed)
{
    con.Open();
}
SqlCommand cmd = new SqlCommand("INSERT INTO
member_master_table(full_name,dob,contact_no,email,state
,city,pincode,full_address,member_id,password,account_st
atus)
values(@full_name,@dob,@contact_no,@email,@state,@city,@
pincode,@full_address,@member_id,@password,@account_stat
us)", con);
cmd.Parameters.AddWithValue("@full_name",
TextBox1.Text.Trim());
cmd.Parameters.AddWithValue("@dob",
TextBox2.Text.Trim());

```

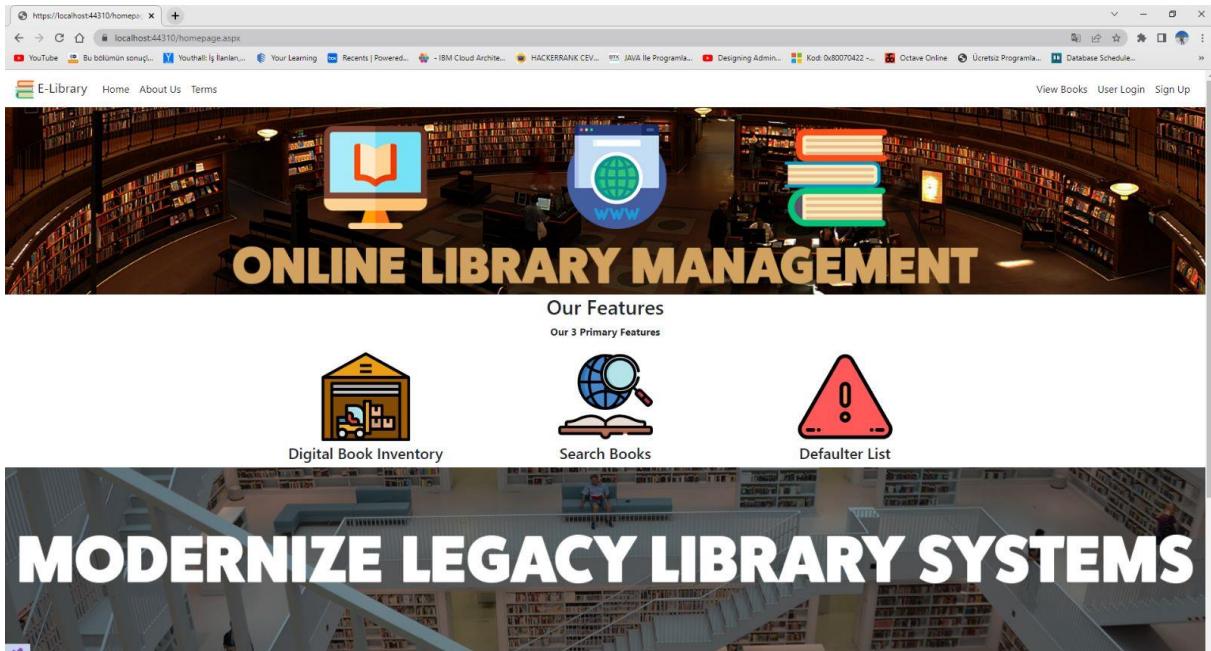
```

        cmd.Parameters.AddWithValue("@contact_no",
        TextBox3.Text.Trim());
        cmd.Parameters.AddWithValue("@email",
        TextBox4.Text.Trim());
        cmd.Parameters.AddWithValue("@state",
        DropDownList1.SelectedItem.Value);
        cmd.Parameters.AddWithValue("@city",
        TextBox6.Text.Trim());
        cmd.Parameters.AddWithValue("@pincode",
        TextBox7.Text.Trim());
        cmd.Parameters.AddWithValue("@full_address",
        TextBox5.Text.Trim());
        cmd.Parameters.AddWithValue("@member_id",
        TextBox8.Text.Trim());
        cmd.Parameters.AddWithValue("@password",
        TextBox9.Text.Trim());
        cmd.Parameters.AddWithValue("@account_status",
        "pending");
        cmd.ExecuteNonQuery();
        con.Close();
        Response.Write("<script>alert('Sign Up Successful. Go to User Login to Login');</script>");
    }
}

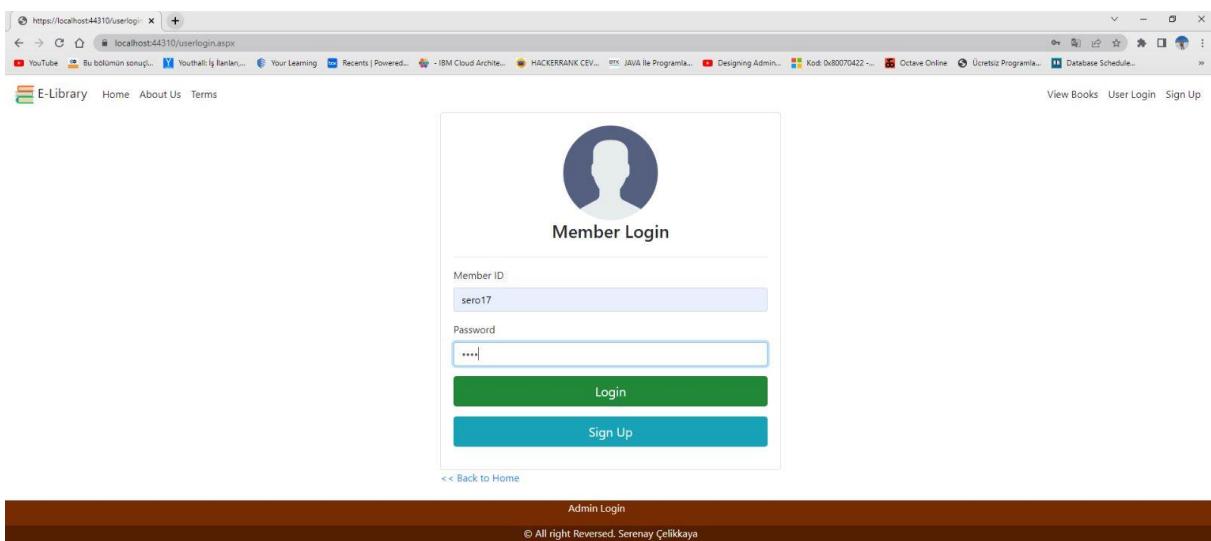
```

full_name	id	contact_no	email	state	city	pincode	full_address	member_id	password	account_status	
Berat Polat	2003-11-16	05000000000	berrat@gmail.com	Kerala	Pathanamthitta	520	Gary Man	berrat00	berrat	23	pending
Hande Celikoglu	2008-11-20	5306452953	hande@gmail.com	Kerala	Kozhikode	520	Fox	hande20	12345		
Serenay Celikoglu	1998-09-01	5077142365	avril@gmail.com	Punjab	Noida	222222	Burur	seri17	1234		pending
Tuncay Salpal	1993-10-18	5655559555	aded@evey.com	Maharashtra	Mumbai	333333	Burur	ts11	123		pending
Zeynep Yildiz	1975-02-20	5236452365	zeynep@gmail.com	Aseam	Dispur	456321	Dispur	zeynep	1999		pending

Şekil 4.3. Kaydolan kullanıcı dbo.member_master_table tablosuna eklendi.



Şekil 4.4. Anasayfada gördüğümüz “User Login” butonuna basıyoruz.



Şekil 4.5. Açılan sayfaya veri tabanına kayıtlı kullanıcılarından herhangi birinin bilgilerini giriyoruz.

Veri tabanına nasıl bağlantı yaptığımızı ve veri okuması yapılpıp eğer o bilgilerde kullanıcı varsa sisteme girişşi sağlanıyor.

```
SqlConnection con =
new
SqlConnection(strcon);
if (con.State == ConnectionState.Closed)
{
    con.Open();
```

```

        }

        SqlCommand cmd = new SqlCommand("select *
from member_master_table where member_id=''
+ TextBox1.Text.Trim() + "' AND password=''
+ TextBox2.Text.Trim() + "'", con);
        SqlDataReader dr = cmd.ExecuteReader();
        if (dr.HasRows)
        {
            while (dr.Read())
            {
                Response.Write("<script>alert('Login
Successfull');</script>");
                Session["username"] =
                    dr.GetValue(8).ToString();
                Session["fullname"] =
                    dr.GetValue(0).ToString();
                Session["role"] = "user";
                Session["status"] =
                    dr.GetValue(10).ToString();
            }
            Response.Redirect("homepage.aspx");
        }
        else
        {
            Response.Write("<script>alert('Invalid
credentials');</script>");
        }
    }
}

```

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure, including the libraryDB database which contains several tables like book_master, member_master, and publisher_master. The central pane displays a query window with the following SQL code:

```

SELECT TOP 1000 [full_name]
      ,[contact_no]
      ,[email]
      ,[state]
      ,[city]
      ,[pincode]
      ,[full_address]
      ,[member_id]
      ,[password]
      ,[account_status]
  FROM [libraryDB].[dbo].[member_master_table]

```

The Results tab shows the output of the query, listing four rows of data:

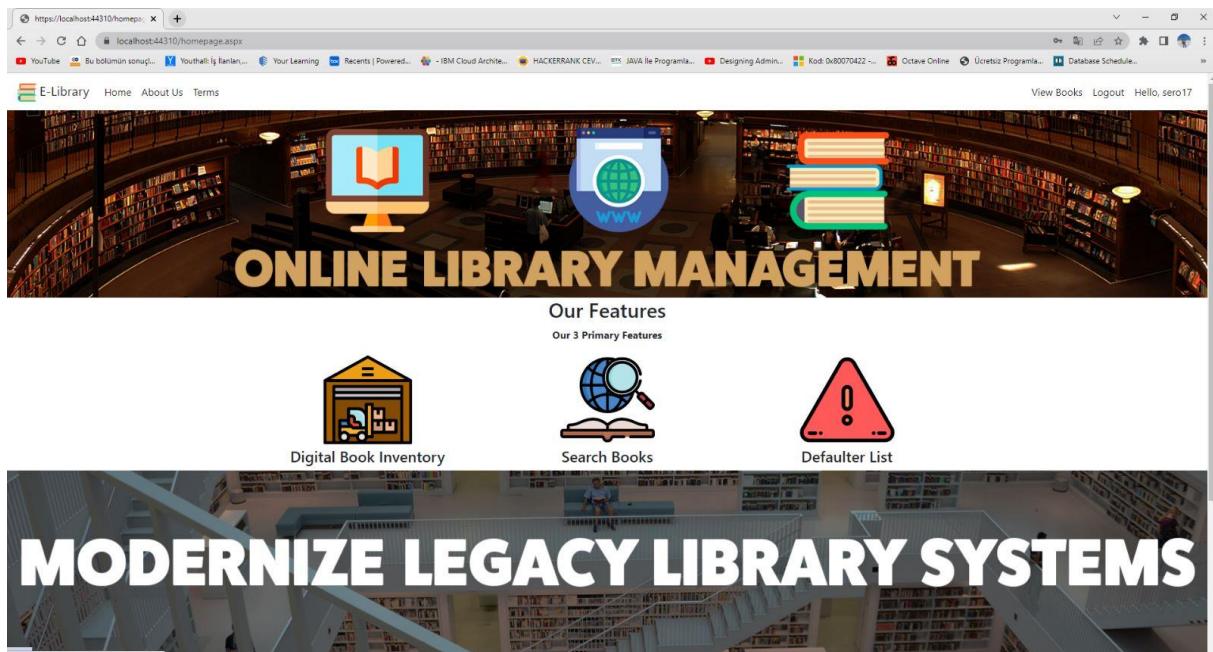
	full_name	dob	contact_no	email	state	city	pincode	full_address	member_id	password	account_status
1	Hanife Colakoglu	2000-01-20	5306452653	hanife@gmail.com	Kerala	Kozi	238541	Kozi	hanife20	12345	Pending
2	Serneye Celikkaya	1999-01-01	5077142655	ad@gnl.com	Punjab	Nagar	222222	Burdur	ser17	1234	Pending
3	Tommy Sekipal	1993-10-18	5656565656	ad@vny.com	Maharashtra	Mumbai	333333	Burdur	ts11	123	Pending
4	Zeynep Yıldız	1975-02-20	5236452365	zeynep@gmail.com	Asean	Dispur	456321	Dispur	zeynepy	1999	Pending

At the bottom of the results pane, it says "Query executed successfully." and shows the status bar with "DESKTOP-BUNLURV (15.0 RTM) DESKTOP-BUNLURV\Serene... libraryDB 00:00:00 | 4 rows".

Şekil 4.6. Veri tabanında dbo.member_master_table tablosunda kayıtlı kullanıcılar

Kullanıcının sisteme girişi şu şekilde sağlanıyor:

```
protected void Button1_Click(object sender, EventArgs e)
{
    try
    {
        SqlConnection con = new SqlConnection(strcon);
        if (con.State == ConnectionState.Closed)
        {
            con.Open();
        }
        SqlCommand cmd = new SqlCommand("select * from
member_master_table where member_id=' " + TextBox1.Text.Trim() +
" ' AND password=' " + TextBox2.Text.Trim() + " ', con);
        SqlDataReader dr = cmd.ExecuteReader();
        if (dr.HasRows)
        {
            while (dr.Read())
            {
                Response.Write("<script>alert('Login Successfull');</script>");
                Session["username"] = dr.GetValue(8).ToString(); // tablodaki
                member_id sütunu yani tablonun 9. Sütundaki bilgiler alınır.
                Session["fullname"] = dr.GetValue(0).ToString();
                Session["role"] = "user";
                Session["status"] = dr.GetValue(10).ToString();
            }
            Response.Redirect("homepage.aspx");
        }
        else
        {
            Response.Write("<script>alert('Invalid
credentials');</script>");
        }
    }
    catch (Exception ex)
    {
    }
}
```



Şekil 4.7. “sero17” kullanıcı adlı kişi giriş yaptıktan sonra sağ üstte “Hello, sero17” yazısını görüyoruz. Kullanıcı, yöneticinin sahip olduğu yetkinliklere(kitap ekleme, verme, yazar ve yayıncı bilgilerini güncelleme) sahip değildir. Kullanıcı sadece kitapları görüntüleyebilir.(“View Books” butonu ile)
Yöneticinin kullanıcı bilgilerini görüntülediği ekrana bakalım.

ID	Name	Account Status	Contact No	Email ID	State	City
hande20	Hande Colakoglu	pending	5306452563	hande@gmail.com	Kerala	Koci
sero17	Serenay Celikkaya	pending	5077142365	ad@gmail.com	Punjab	Niagor
ts11	Tanmay Sakpal	pending	5665656565	adad@exy.com	Maharashtra	Mumbai
zeynep	Zeynep Yildiz	pending	5236452365	zeynep@gmail.com	Assam	Dispur

Şekil 4.8. Yönetici “Member Management” butonuna tıklayarak bu sayfaya ulaşır ve member ID yazılıp tıkla istediği kullanıcının bilgilerini görecektir ve hesap durumunu(aktif,bekliyor,aktif değil) güncelleyebilir, kullanıcıyı sistemden silebilir.
Yöneticinin kullanıcıyı silebilmesi için:

```

SqlCommand cmd = new
SqlCommand("DELETE
from
member_master_table
WHERE member_id='"
+ TextBox1.Text.Trim()
+ "'", con);
cmd.ExecuteNonQuery();
con.Close();
Response.Write("<script>alert('Member Deleted
Successfully');</script>");
clearForm();
GridView1.DataBind();

```

The screenshot shows a web application interface. On the left, there is a 'Member Details' form with fields for Member ID (sero17), Full Name (Serenay Celikkaya), Account Status (pending), DOB (1999-09-01), Contact No (5077142365), Email ID (ad@gmail.com), State (Punjab), City (Niagor), and Pin Code (222222). It also has a 'Full Postal Address' field containing 'Burdur'. At the bottom of this form is a red button labeled 'Delete User Permanently'. On the right, there is a 'Member List' table with the following data:

ID	Name	Account Status	Contact No	Email ID	State	City
hande20	Hande Colakoglu	pending	5306452563	hande@gmail.com	Kerala	Kochi
sero17	Serenay Celikkaya	pending	5077142365	ad@gmail.com	Punjab	Niagor
ts11	Tanmay Sakpal	pending	5665656565	adad@exy.com	Maharashtra	Mumbai
zeynepy	Zeynep Yildiz	pending	5236452365	zeynep@gmail.com	Assam	Dispur

Below the table, it says 'Showing 1 to 4 of 4 entries'. At the bottom of the page, there is a navigation bar with links for Author Management, Publisher Management, Book Inventory, Book Issuing, Member Management, and a copyright notice: '© All right Reversed. Serenay Celikkaya'.

Şekil 4.9. Kullanıcı bilgileri getirildi.

Kullanıcı bilgilerini şu şekilde çağrııyorum:

```

SqlCommand cmd = new
SqlCommand("select *
from
member_master_table
where member_id='"
+ TextBox1.Text.Trim()
+ "'", con);

SqlDataReader dr = cmd.ExecuteReader();
if (dr.HasRows)
{
    while (dr.Read())
    {

```

```

    TextBox2.Text = dr.GetValue(0).ToString();
    //full name of user
    TextBox7.Text = dr.GetValue(10).ToString();
    //account status of user
    TextBox8.Text = dr.GetValue(1).ToString();
    TextBox3.Text = dr.GetValue(2).ToString();
    TextBox4.Text = dr.GetValue(3).ToString();
    TextBox9.Text = dr.GetValue(4).ToString();
    TextBox10.Text = dr.GetValue(5).ToString();
    TextBox11.Text = dr.GetValue(6).ToString();
    TextBox6.Text = dr.GetValue(7).ToString();
}
}
else
{
    Response.Write("<script>alert('Invalid
credentials');</script>");
}

```

5 – Çoklu seçim, ekleme ve resim dosyası yükleme ile kitap seçmenin detaylarını güncelleme ve ImageView ile Özel Gridview kullanarak projeyi tamamlama

Çoklu seçim, ekleme ve resim dosyası yükleme yapılarak kitap seçmenin detayları 3 numaralı maddede görselleriyle beraber anlatılmıştır. Örneğin Şekil 3.20.’de resim dosyası yükleme, 3.21.’de kitap türü, yayıncı adı ve kitabıng hangi dilde yazıldığını seçmek için çoklu seçim, 3.24.’te sayfada eklenen kitapları görme ve arama kısmı gösterilip anlatılmıştır.