

ANKARA ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



BLM 4531 PROJE RAPORU

Kütüphane Otomasyonu

Berkay Akparlar

19290207

Enver BAĞCI

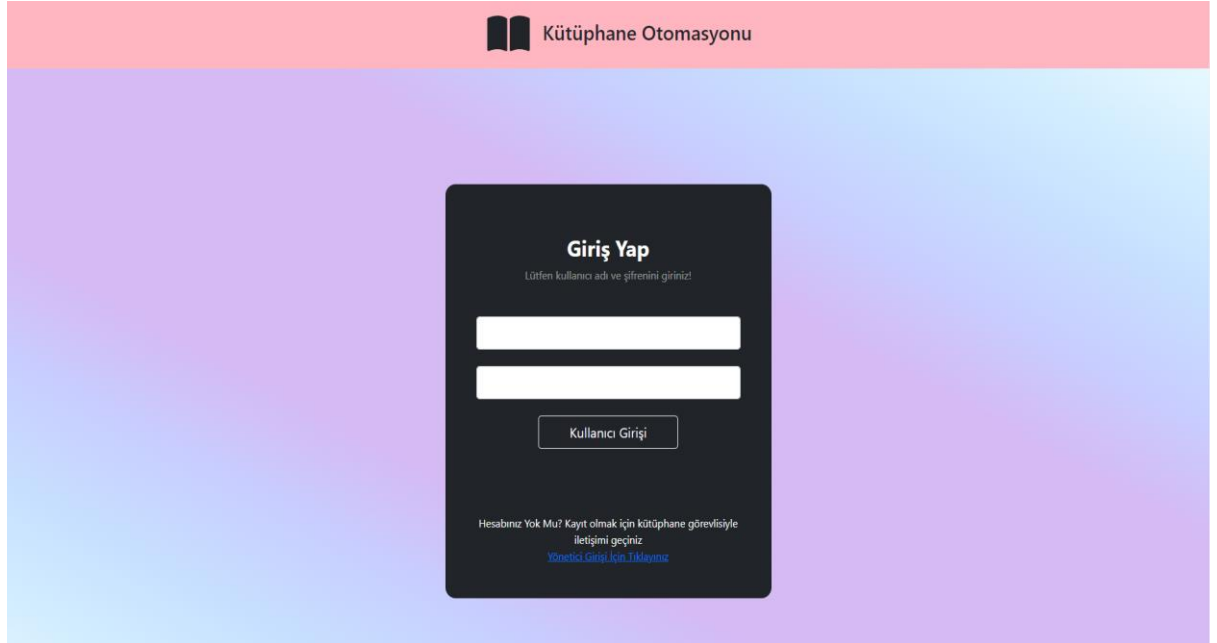
26.01.2023

Projenin Tanıtımı

Proje kütüphaneyi yönetmek için gerekli temel özellikleri içinde bulunduruyor. Bunlar başlıca kitap ekleme, kitap güncelleme, kitapları belli durumlara göre filtrelemek, kitap emanet alma sistemi, beğeni listesi oluşturma gibi özelliklerdir.

Sitenin Görünümü

Sayfaya ilk girildiğinde giriş yapma ekranı karşımıza çıkıyor.



Kütüphane Otomasyonu

Giriş Yap

Lütfen kullanıcı adı ve şifrenizi giriniz!

Kullanıcı Girişi

Hesabınız Yok Mu? Kayıt olmak için kütüphane görevlisiyle iletişimi geçiniz

[Yönetici Giriş İçin Tıklayınız](#)

Burada kullanıcı giriş ekranı görünmektedir. Eğer yönetici girişi yapmak istersek sayfanın altında bulunun bağlantı üzerinden giriş sayfasına yönlendirme yapılmaktadır.

Admin Paneli

Kitap Id	Kitap Foto	Kitap Ad	Kitap Yazar	Kitap Tür	İşlemler
1		Son Kuşlar	Reşat Nuri Gültekin	Tarih	Sil Güncelle Detay
2		Başlangıç	Dan Brown	Korku-Gerilim	Sil Güncelle Detay
3		Kayıp Sembol	Dan Brown	Korku-Gerilim	Sil Güncelle Detay
4		Yüzüklerin Efendisi - I - Yüzük Kardeşliği	John Ronald Reuel Tolkien	Fantastik	Sil Güncelle Detay
5		Çalıkuşu	Reşat Nuri Gültekin	Romantik	Sil Güncelle Detay

Yeni Kitap Kaydı

Admin paneline giriş yapıldığında kitapların listelendiği sayfa görülmektedir. Bu sayfada kitapları silme, güncelleme, yeni kitap ekleme işlemleri yapılabilmektedir.

Kitap Silme Ekranı

"Son Kuşlar" Kitabını Silme İşlemini Onaylıyor Musunuz

[Evet](#) [Hayır](#)

Sil butonuna tıkladığımızda bu şekilde popup açılmaktadır.

Kitap Güncelleme Sayfası

KitapAd	Son Kuşlar
KitapYazar	Reşat Nuri Gültekin
KitapTur	Tarih
SayfaSayisi	110
BaskiSayisi	10
KitapAdeti	10
KitapFotoUrl	https://img.kitapyurdu.com/v1/getImage/fn:11409634/wh:true/wi:220
KitapOzet	"Söz vermiştim kendi kendime: Yazı bile yazmayacaktım. Yazı yazmak da, bir hırstan başka ne idi? Burada namuslu insa
Güncelle	

Güncelle sayfası bu şekildedir. Seçtiğimiz kitabın bilgileri buraya taşınmakta. Kitap Tür ve Yazar dropdownlist ile seçilmektedir.

Kitap Ekleme Sayfası

Kitap Ad	
Yazar Adı	Yazar Seçiniz
Kitap Türü	Kitap Türünü Seçiniz
Sayfa Sayısı	
Baskı Sayısı	
Kitap Adeti	
Kitap Kapak Resmi Url	
Kitap Özeti	
Kaydet	



Kitap Ekleme sayfasında ür ve Yazar dropdownlist ile seçilmektedir diğer bilgiler elle girilerek kitap ekleme işi tamamlanır.

Tür Id	Tür Adı	İşlemler
1	Korku-Gerilim	<input type="button" value="Sil"/> <input type="button" value="Güncelle"/>
2	Polisiye	<input type="button" value="Sil"/> <input type="button" value="Güncelle"/>
3	Romantik	<input type="button" value="Sil"/> <input type="button" value="Güncelle"/>
4	Otobiyografi	<input type="button" value="Sil"/> <input type="button" value="Güncelle"/>
5	Bilim Kurgu	<input type="button" value="Sil"/> <input type="button" value="Güncelle"/>
6	Tarih	<input type="button" value="Sil"/> <input type="button" value="Güncelle"/>
7	Felsefe	<input type="button" value="Sil"/> <input type="button" value="Güncelle"/>
8	Kişisel Gelişim	<input type="button" value="Sil"/> <input type="button" value="Güncelle"/>
9	Fantastik	<input type="button" value="Sil"/> <input type="button" value="Güncelle"/>
11	denemeeee	<input type="button" value="Sil"/> <input type="button" value="Güncelle"/>

Kitapların kategorileri bu sayfada görülmektedir.

Tür Ekleme Sayfası

Tür Adı

Yeni Tür ekleme işlemi popup üzerinden bu şekilde olmaktadır.

Tür Güncelleme Ekranı

Tür Adı

Otobiyografi

Güncelle

Çıkış

Kategori güncelleme butonuna basıldığında seçtiğimiz kategori bu şekilde gelmektedir. Üzerinde değişiklik yapıp güncelleme işlemi tamamlanmaktadır.






Tür Silme Ekranı

"Otobiyografi" Kulübünü Silme İşlemini Onaylıyor Musunuz

Evet

Hayır

Kategori silme işlemi bu şekilde olmaktadır.

Kitap Id	Kitap Fotoğrafl	Kitap Adı	Kitap Adeti	İşlemler
1		Son Kuşlar	10	<input type="button" value="Yeni Sevkiyat"/>
2		Başlangıç	1	<input type="button" value="Yeni Sevkiyat"/>
3		Kayıp Sembol	2	<input type="button" value="Yeni Sevkiyat"/>
4		Yüzüklerin Efendisi - I - Yüzük Kardeşliği	5	<input type="button" value="Yeni Sevkiyat"/>
5		Çalılıkusu	2	<input type="button" value="Yeni Sevkiyat"/>

1

Kütüphanede bulunan kitapların stok adetleri bu sayfada görüntülenmektedir.

Kitap Sevkiyat Ekranı

×

Son Kuşlar

Yeni Gelen Kitap Adeti

Kütüphaneye elimizde olan kitaplardan tekrar geldiğinde yeni gelen kitap adeti burdan girilerek stok adeti güncellenmektedir.

Emanet Id	Kitap Foto	Kitap Ad	Emanet Alan	Emanet Alınan Tarih	Teslim Tarihi	Durumu	İşlemler
1		Son Kuşlar	Berkay Akparlar	19.12.2022 00:00:00	21.12.2022 00:00:00	Süresi Gelmedi	<input type="button" value="Kitap Teslimi"/> <input type="button" value="Süre Uzatma"/>

1

Bu sayfada emanet alınan kitaplar listelenmektedir. Burada kitap ismi, emanet alan kullanıcı, emanet alınan ve teslim edilme tarihi görülmektedir.

Teslim Süresi Uzatma Ekranı

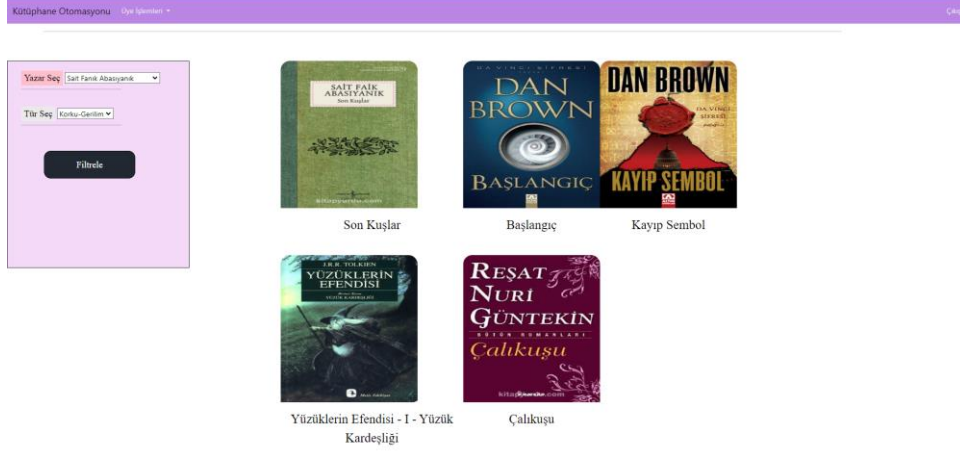
Eski Teslim Tarihi: 21.12.2022 00:00:00

Yeni Teslim Tarihini Giriniz

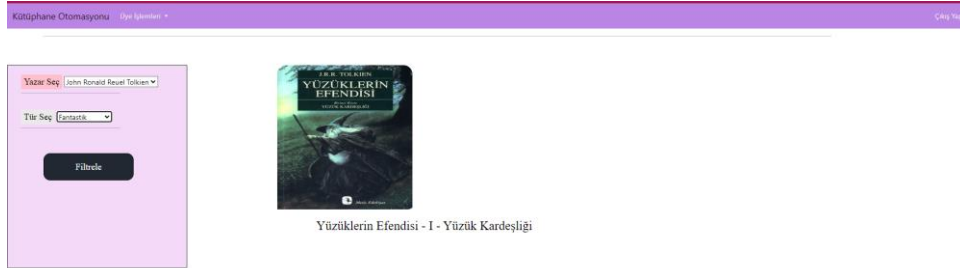
Süre uzatma işleminde yeni girdiğimiz tarih ile kitabın teslim tarihi güncellenmiş olmaktadır.

Kullanıcı Ekranları

Kullanıcı sisteme giriş yaptığinde kitapların listelendiği sayfayı görmektedir.



Sol trafta bulunan filtreleme seçenekleri üzerinden kategori ve yazarlar filtrelenir.



Yukarıdaki fotoğrafta filtreleme örneği görülmektedir.

Kod Örnekleri

```
namespace Kutuphane.Controllers
{
    // 0 references
    public class AdminKitapListeController : Controller
    {
        // GET: AdminKitapListe
        LIBRARYEntities0 db = new LIBRARYEntities0();
        // 0 references
        public ActionResult Index(string search, int sayfa = 1)
        {
            return View(
                db.Kitaplar.Where(x => x.KitapAd.Contains(
                    search) || x.Yazarlar.YazarAd.Contains(search) || x.Turler.TurAd.Contains(search) || search == null
                ).ToList().ToPagedList(sayfa, 5));
        }

        [HttpGet]
        // 0 references
        public ActionResult YeniKitap()
        {
            List<SelectListItem> yazarListesi = (from i in db.Yazarlar.ToList()
                                                select new SelectListItem
                                                {
                                                    Text = i.YazarAd,
                                                    Value = i.YazarId.ToString()
                                                }).ToList();

            ViewBag.yazar = yazarListesi;

            List<SelectListItem> turListesi = (from i in db.Turler.ToList()
                                                select new SelectListItem
                                                {
                                                    Text = i.TurAd,
                                                    Value = i.TurId.ToString()
                                                }).ToList();

            ViewBag.tur = turListesi;
        }
    }
}
```

Yukarıdaki kod örneğinde admin panelinde kitapların listelendiği sayfada arama özelliğini sağlayan serachar için gerekli kodlar görünmektedir. Ayrıca kitap ekleme sayfasında yazar ve türleri dropdownlist ile seçmemizi sağlayan listeler oluşturulmuştur.

```

0 references
public ActionResult SilKitap(int id)
{
    Kitaplar kitap = db.Kitaplar.Where(x => x.KitapId == id).FirstOrDefault();
    db.Kitaplar.Remove(kitap);
    db.SaveChanges();
    TempData["Message"] = " Kitap Başarıyla Silindi.";
    return RedirectToAction("Index");
}

[HttpGet]
0 references
public ActionResult GetirKitap(int id)
{
    List<SelectListItem> yazarListesi = (from i in db.Yazarlar.ToList()
                                        select new SelectListItem
                                        {
                                            Text = i.YazarAd,
                                            Value = i.YazarId.ToString()
                                        }).ToList();

    ViewBag.yazar = yazarListesi;

    List<SelectListItem> turListesi = (from i in db.Turler.ToList()
                                       select new SelectListItem
                                       {
                                           Text = i.TurAd,
                                           Value = i.TurId.ToString()
                                       }).ToList();

    ViewBag.tur = turListesi;

    var kitap = db.Kitaplar.Find(id);
    return View("GetirKitap", kitap);
}

```

Seçilen kitabı silmemizi sağlayan ve güncelleme için seçilen kitabın bilgilerini getiren controllerlar şekildeki gibidir.

```

[HttpGet]
0 references
public ActionResult GuncelleKitap()

[HttpPost]
0 references
public ActionResult GuncelleKitap(KitapModel model)
{
    Kitaplar kitapUpdate = db.Kitaplar.Where(x=>x.KitapId == model.KitapId).FirstOrDefault();

    var yazar = db.Yazarlar.Where(m => m.YazarId == model.KitapYazar).FirstOrDefault();
    var tur = db.Turler.Where(m => m.TurId == model.KitapTur).FirstOrDefault();

    kitapUpdate.Yazarlar = yazar;
    kitapUpdate.Turler = tur;

    kitapUpdate.KitapAd = model.KitapAd;
    kitapUpdate.SayfaSayisi = model.SayfaSayisi;
    kitapUpdate.BaskiSayisi = model.BaskiSayisi;
    kitapUpdate.KitapOzet = model.KitapOzet;
    kitapUpdate.KitapAdeti = model.KitapAdeti;
    kitapUpdate.KitapAdeti = model.KitapAdeti;
    kitapUpdate.KitapFotoUrl = model.KitapFotoUrl;
    db.SaveChanges();
    return RedirectToAction("Index");
}

```

Yukarıdaki fotoğrafta görülen kod ile kitap güncelleme işlemi yapılmaktadır. Yazar ve tür görüldüğü üzere dropdownlist ile seçilmektedir.

```

[HttpPost]
[ValidateAntiForgeryToken]
0 references
public ActionResult SilTur(int id)
{
    Turler kulup = db.Turler.Where(x => x.TurId == id).FirstOrDefault();
    db.Turler.Remove(kulup);
    db.SaveChanges();
    TempData["Message"] = " Tür Başarıyla Silindi.";
    return RedirectToAction("Index");
}

```

```

[HttpGet]
0 references
public ActionResult GetirTur()...

```

```

[HttpPost]
0 references
public ActionResult GetirTur(int id)
{
    var turName = db.Turler.Find(id);
    return View("GetirTur", turName);
}

```

```

[HttpGet]
0 references
public ActionResult GuncelleTur()...

```

```

[HttpPost]
0 references
public ActionResult GuncelleTur(TurModel model)
{
    Turler turUpdate = db.Turler.Where(x => x.TurId == model.TurId).FirstOrDefault();
    turUpdate.TurAd = model.TurAd;
    db.SaveChanges();
    TempData["Message"] = " Tür Başarıyla Güncellendi.";

    return RedirectToAction("Index");
}

```

Bu sayfada kategori silme, güncelleme işlemi için kategoriye, getirme ve işlem yapıldıktan sonra güncelleme işlemi yapılmasını sağlayan controllerlar görülmektedir.

```

[HttpPost]
[ValidateAntiForgeryToken]
0 references
public ActionResult Index(Kullanici US)
{
    var checkLogin = db.Kullanicilar.Where(x => x.KullaniciAd.Equals(US.KullaniciAd)
    && x.KullaniciSifre.Equals(US.KullaniciSifre)).FirstOrDefault();

    if (checkLogin != null)
    {
        Session["USERTCSS"] = US.KullaniciAd.ToString();
        FormsAuthentication.SetAuthCookie(US.KullaniciAd, false);
        return RedirectToAction("Index", "Home");
    }
    else
    {
        ViewBag.Notification = "Yanlış kullanıcı adı veya şifre";
    }
    return View();
}

[HttpGet]
0 references
public ActionResult AdminLogin()
{
    return View();
}

[HttpPost]
0 references
public ActionResult AdminLogin(Adminler adminler)
{
    var checkLogin = db.Adminler.Where(x => x.AdminKullaniciId.Equals(adminler.AdminKullaniciId)
    && x.AdminSifre.Equals(adminler.AdminSifre)).FirstOrDefault();

    if (checkLogin != null)
    {
        Session["USERTCSS"] = adminler.AdminKullaniciId.ToString();
        FormsAuthentication.SetAuthCookie(adminler.AdminKullaniciId, false);

        return RedirectToAction("Index", "AdminKitapListe");
    }
    else
    {
        ViewBag.Notification = "Yanlış kullanıcı adı veya şifre";
    }
    return View();
}

```

Login işlemlerinin gerçekleştiği kısım buradadır. Üstteki kısımda kullanıcı girişi, alttaki controllerda ise admin girişi yapılmaktadır.

Genel olarak controllerlar bu şekildedir. Hepsini tanıtmam raporu fazla uzatacağından önemki controlleri tanıttım. Şimdi de viewları tanıtmaya geçiyorum.

```

@using (Html.BeginForm("Index", "AdminKitapListe", FormMethod.Get))
{
    <div style="text-align: end; margin-top: 2%; margin-right: 5% ">
        @Html.TextBox("search")
        <input type="submit" value="Search" class="btn btn-info" />
    </div>
}
<div style="margin-right:5%;margin-left:5%; margin-top:2%;">

    <table class="table table-bordered">
        <thead>
            <tr style="text-align:center;">
                <th>Kitap Id</th>
                <th>Kitap Foto</th>
                <th>Kitap Ad</th>
                <th>Kitap Yazar</th>
                <th>Kitap Tür</th>
                <th>İşlemler</th>
            </tr>
        </thead>
        <tbody>
            @if (Model.Count() == 0)
            {
                <tr>
                    <td>Kayıt Bulunamadı</td>
                </tr>
            }
            else
            {
                foreach (var kitap in Model)
                {
                    <tr style="text-align: center; ">
                        <td style="vertical-align:middle;">@kitap.KitapId</td>
                        <td>  </td>
                        <td style="vertical-align: middle">@kitap.KitapAd</td>
                        <td style="vertical-align:middle;">@kitap.Yazarlar.YazarAd</td>
                        <td style="vertical-align:middle;">@kitap.Turler.TurAd</td>
                        <td style="vertical-align:middle;">
                            <a id="btnDelete" class="btn btn-danger btn-sm" data-toggle="modal" data-target="#DeleteModal-@kitap.KitapId">Sil</a>
                            <a href="/AdminKitapListe/GetirKitap/@kitap.KitapId" class="btn btn-info">Güncelle</a>
                            <a href="/AdminKitapListe/DetayKitap/@kitap.KitapId" class="btn btn-primary">Detay</a>
                        </td>
                    </tr>
                }
            }
        </tbody>
    </table>

    @using (Html.BeginForm("SilKitap", "AdminKitapListe", new { id = kitap.KitapId }, FormMethod.Post, null))
    {
        @Html.AntiForgeryToken()
        <div class="modal" tabindex="-1" role="dialog" id="DeleteModal-@kitap.KitapId">
            <div class="modal-dialog" role="document">
                <div class="modal-content">
                    <div class="modal-header">
                        <h4 class="modal-title">Kitap Silme Ekranı</h4>
                        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                            <span aria-hidden="true">&times;</span>
                        </button>
                    </div>
                    <div class="modal-body">
                        @kitap.KitapAd Kitabını Silme İşlemini Onaylıyor Musunuz
                    </div>
                    <div class="modal-footer">
                        <button type="submit" class="btn btn-danger">Evet</button>
                        <button type="button" class="btn btn-secondary" data-dismiss="modal">Hayır</button>
                    </div>
                </div>
            </div>
        </div>
    }
}

```

Bu sayfalarda admin panelinde kitapların listelendiği sayfanın kodları görünmektedir.

Silme işlemi için kullanılan popup burada görünmektedir.

```

using (Html.BeginForm("GuncelleKitap", "AdminKitapListe", new { @class = "form-group" }, FormMethod.Post, null))
{
    <div style="text-align:center; justify-content:center;">

        @Html.HiddenFor(m=>m.KitapId)

        <div style="margin-top: 30px; margin-right: 10%; margin-left: 10%">
            @Html.LabelFor(m => m.KitapAd)
            @Html.TextBoxFor(m => m.KitapAd, new { @class = "form-control" })
        </div>

        <div style="margin-top: 30px; margin-right: 10%; margin-left: 10%">
            @Html.LabelFor(m => m.KitapYazar)
            @Html.DropDownListFor(m => m.KitapYazar, (List<SelectListItem>)ViewBag.yazar, "Yazar Seçiniz", new { @class = "form-control" })
        </div>

        <div style="margin-top: 30px; margin-right: 10%; margin-left: 10%">
            @Html.LabelFor(m => m.KitapTur)
            @Html.DropDownListFor(n => n.KitapTur, (List<SelectListItem>)ViewBag.tur, "Kitap Türünü Seçiniz", new { @class = "form-control" })
        </div>

        <div style="margin-top: 30px; margin-right: 10%; margin-left: 10%">
            @Html.LabelFor(m => m.SayfaSayisi)
            @Html.TextBoxFor(m => m.SayfaSayisi, new { @class = "form-control" })
        </div>

        <div style="margin-top: 30px; margin-right: 10%; margin-left: 10%">
            @Html.LabelFor(m => m.BaskiSayisi)
            @Html.TextBoxFor(m => m.BaskiSayisi, new { @class = "form-control" })
        </div>

        <div style="margin-top: 30px; margin-right: 10%; margin-left: 10%">
            @Html.LabelFor(m => m.KitapAdeti)
            @Html.TextBoxFor(m => m.KitapAdeti, new { @class = "form-control" })
        </div>

        <div style="margin-top: 30px; margin-right: 10%; margin-left: 10%">
            @Html.LabelFor(m => m.KitapFotoUrl)
            @Html.TextBoxFor(m => m.KitapFotoUrl, new { @class = "form-control" })
        </div>

        <div style="margin-top: 30px; margin-right: 10%; margin-left: 10%">
            @Html.LabelFor(m => m.KitapOzet)
            @Html.TextBoxFor(m => m.KitapOzet, new { @class = "form-control ozet" })
        </div>

        <div style="margin-top:10px">
            <button class="button btn btn-info">Güncelle</button>
        </div>
    </div>
}

```

Admin tarafından kitapların güncelleme işleminin gerçekleştiği view sayfasının kodları bu şekildedir.

```

@using (Html.BeginForm("Index", "UserKitapListe", FormMethod.Post))
{
    <div class=" clearfix float-my-children ">
        <div class="filter">
            <div class="dropdown">
                <button class=" " style="margin-left:30px;margin-top:20px; color: black; border: none;
font-weight: 500;font-size:20px; font-family: Poppins; background-color: pink;
" type="button" data-bs-toggle="dropdown" aria-expanded="false">
                    Yazar Seç
                </button>
                <select name="yazarfiltre">
                    @foreach (var item in Model.Types)
                    {
                        <option value="@item.Text">@item.Text</option>
                    }
                </select>
                <ul class="dropdown-menu">
                    @foreach (var x in Model.Types)
                    {
                        <li><div class="dropdown-item" style="color: black;
font-weight: 500; font-family: Poppins;">@x</div></li>
                    }
                </ul>
            </div>
            <hr class="solid2">
            <div class="dropdown">
                <button class=" " style="margin-left:30px;margin-top:20px; color: black; border: none;
font-weight: 500;font-size:20px; font-family: Poppins; background-color: #EBEBEB;
" type="button" data-bs-toggle="dropdown" aria-expanded="false">
                    Tür Seç
                </button>
                <select name="turfiltre">
                    @foreach (var item in Model.Types2)
                    {
                        <option value="@item.Text">@item.Text</option>
                    }
                </select>
                <ul class="dropdown-menu">
                    @foreach (var x in Model.Types2)
                    {
                        <li><div class="dropdown-item" style="color: black;
font-weight: 500; font-family: Poppins;">@x</div></li>
                    }
                </ul>
            </div>
            <hr class="solid2">
            <button type="submit" style="height: 60px; width: 200px; background-color: #222831;
border-radius: 15px; margin-top:40px; margin-left: 80px; font-size:20px;font-weight: 500;
font-family: Poppins;color:white ">Filtrele</button>
        </div>
    </div>

    <div class="grid-container">
        @foreach (var x in Model.Kitaplars)
        {
            <div style="display:@(x.KitapAd==null?"none" : "block");">
                <a href="/KitapInfo?id=@x.KitapId" class="grid-item" style="text-decoration:none;">
                    
                    <div style="font-family: Poppins; color: black; font-size: 28px; font-weight: 500;margin-top:15px;">
                        @x.KitapAd
                    </div>
                </a>
            </div>
        }
    </div>
}

```

Kullanıcı tarafında kitapların listelenmesi ve yazar, tür filtreleneşinin görüldüğü view bu şekildedir.