

ANKARA ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



BLM4531 Ağ Tabanlı Teknolojiler ve Uygulamaları

Rapor

TimeZona

Mesut ÖZLAHLAN

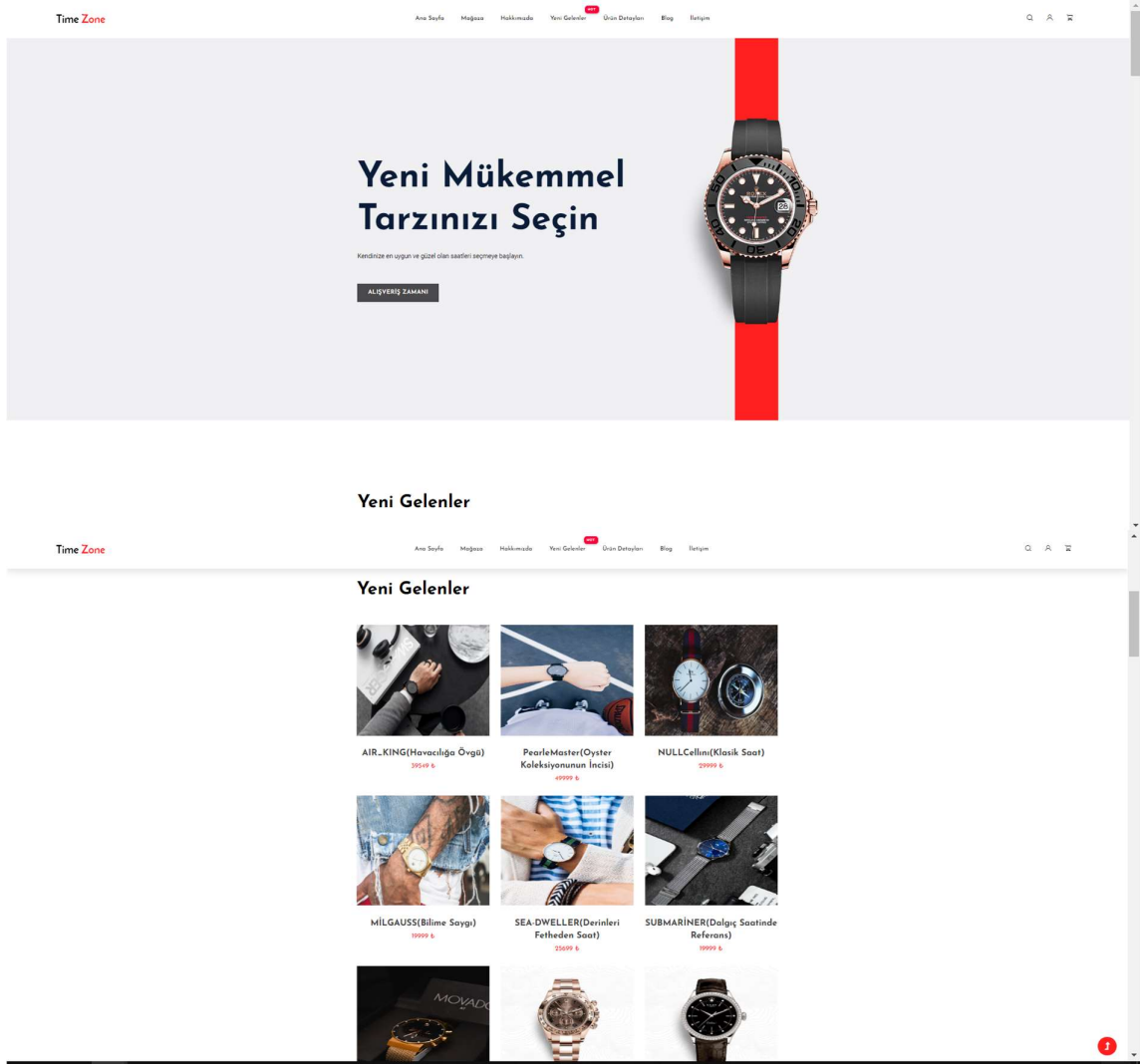
18290051

Ocak 2023

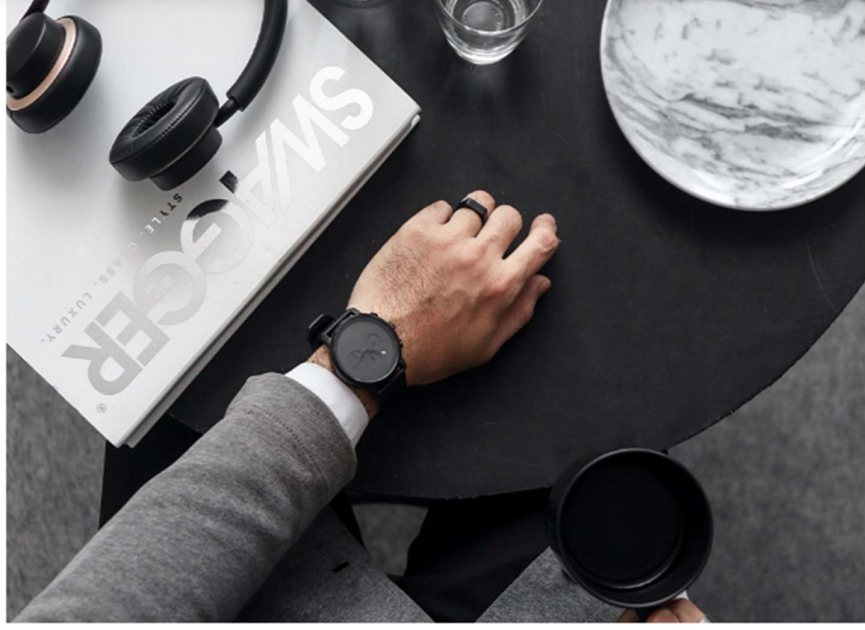
BLM4531 Ağ Tabanlı Teknolojiler ve Uygulamalar dersi kapsamında Microsoft tarafından geliştirilen Asp.Net teknolojisi ile “Timezona” isminde veritabanı içeren bir web uygulaması geliştirilmiştir. Proje konusu olan Timezona, saatlerin satıldığı bir e-ticaret sitesidir.

1. Önyüz (Frontend)

Geliştirilen web uygulamasının önyüz tasarımları; uygulamayı geliştirmeden önce tasarlanan prototipe göre tasarlanmıştır.



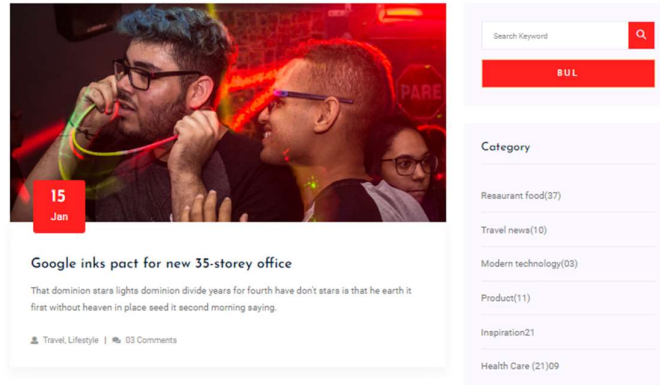
Ana sayfada saatler ve bunların fiyat bilgisi bulunur.



AIR_KING(HavacılığaÖvgü)



Air-King, saatleri gösteren büyük 3, 6 ve 9 rakamlarının bulunduğu çarpıcı siyah kadranı ve kolay görülür bir dakika ölçeğini bir araya getirerek seyir süresini takip etmeyi sağlar. Kadranın üzerindeki Air-King logosu, model için 1950'lerde özel olarak tasarlanmış karakterlerle yazılmıştır. Air-King, Rolex'in 1930'lu yıllarda altın çağını yaşayan havacılık dünyasıyla kurduğu ayrıcalıklı ilişkinin somut örneğidir. Bu dönemde havacılık alanında görülen baş döndürücü ilerlemeler, insanlığın gökleri fethetmesini kolaylaştırarak uzun mesafe uçuşlarının da önünü açmıştır.

Ürün detayları sayfasında saatlerin özellikleri API den çekilir.



Blog kısmında API den çekilen haberler bulunmaktadır.

yapılır, kayıt yoksa hesap oluşturun bölümünden yeni bir kullanıcı oluşturulup veri tabanına kaydedilir.

Product	Price	Quantity	Total
	\$360.00	1	\$720.00
	\$360.00	1	\$720.00

Update CartClose Coupon

Subtotal

\$2160.00

Shipping

Flat Rate: \$5.00

Free Shipping

Flat Rate: \$10.00

Local Delivery: \$2.00

Calculate Shipping

Türkiye

Select a State

Posta kodu

Update Details

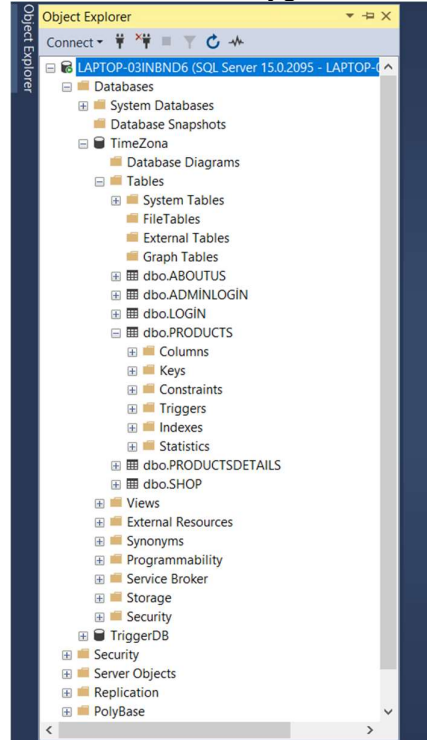
Continue Shopping

Proceed To Checkout

Sepetim bölümünde almak istediğimiz ürünleri ekleyip sipariş verebildiğimiz ödeme ekranımızdır.

2. Veritabanı

Uygulamada kayıt altında tutulan veriler için gerekli veritabanı işlemleri MSSQL kullanılarak yapılmıştır. “TimeZona” isminde bir veritabanı oluşturulmuş ve bu veritabanı altında uygulamada ihtiyaç duyulan tablolar oluşturulmuştur.



ID	PHOTO	NAME	PRICE	QUEUE	ACTIVE
1	gallery1.pn...	AIR_KING(H...	39549	100	1
2	gallery2.pn...	PearleMaste...	49999	150	1
3	gallery3.pn...	NULLCellini(...	29999	200	1
4	gallery4.pn...	MILGAUSS(...	19999	250	1
5	new_produc...	SEA-DWELL...	25699	300	1
6	new_produc...	SUBMARIN...	19999	350	1
8	new_produc...	YACHT-MA...	45999	400	1
9	popular1.pn...	COSMOGRA...	25699	450	1
10	popular2.pn...	DAY-DATE(...	59999	500	1
11	popular3.pn...	GMT-MAST...	19999	550	1
12	popular4.pn...	SKY-DWELL...	24999	600	1
13	popular5.pn...	OYSTER PER...	12999	650	1
15	popular6.pn...	DATEJUST(K...	11999	700	1
16	sectin_bg1...	EXPLORER(...	19999	750	1
* NULL	NULL	NULL	NULL	NULL	NULL

PRODUCTS	
ID	
PHOTO	
NAME	
PRICE	
QUEUE	
ACTIVE	

ADMINLOGIN	
ID	
NAMESURNAME	
PASSWORD	
ROLE	

LOGIN	
ID	
NAMESURNAME	
PASSWORD	

PRODUCTSDetails	
ID	
PHOTO	
NAME	
CONTENTS	
QUEUE	
ACTIVE	

SHOP	
ID	
PHOTO	
NAME	
PRICE	

ABOUTUS	
ID	
HEADER	
SUBJECT	
ACTIVE	
[ORDER]	

3. Arkayüz (Backend)

Arkayüzde yapılan kodlama sürecinde ilk başta uygulamanın veritabanı ile bağlantısını sağlayabilmek için gerekli kodlamalar yapılmıştır.

```
<connectionStrings>
  <add name="TimeZonaEntities" connectionString="metadata=res://*/Models.Model.csdl|res://*/Models.Model.ssdl|res://*/Models.Model.msl;provider=System.Data.SqlClient;provider co
</connectionStrings>
```

```
public ActionResult GLogin(LOGIN kullanıcıFormu)
{
    if (!ModelState.IsValid)
    {
        return View("index", kullanıcıFormu);
    }
    using (TimeZonaEntities db = new TimeZonaEntities())
    {
        var kullanıcıVarı = db.LOGIN.FirstOrDefault(
            x => x.NAMESURNAME == kullanıcıFormu.NAMESURNAME && x.PASSWORD == kullanıcıFormu.PASSWORD);
        if (kullanıcıVarı != null)
        {
            FormsAuthentication.SetAuthCookie(kullanıcıVarı.NAMESURNAME, kullanıcıFormu.BeniHatırla);
            return RedirectToAction("/index", "magaza");
        }
        ViewBag.Hata = "Kullanıcı Adı veya Şifre Hatalı";
        return View("Giriş");
    }
}
```

Giriş ekranı kodları(şifreleme için MD5 kullanıldı)


```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.Mvc;
6  using System.Web.Security;
7  using TimeZona.Models;
8
9
10 namespace TimeZona.Controllers
11 {
12     public class DefaultController : Controller
13     {
14
15         // GET: Default
16         [Route("anasayfa")]
17         public ActionResult Index()
18         {
19             using (TimeZonaEntities db = new TimeZonaEntities())
20             {
21                 var model = db.PRODUCTS.ToList();
22                 return View(model);
23             }
24         }
25
26         [Route("magaza")]
27         public ActionResult Magaza()
28         {
29             using (TimeZonaEntities db = new TimeZonaEntities())
30             {
31                 var model = db.PRODUCTS.ToList();
32                 return View(model);
33             }
34         }
35
36         [Route("hakkimizda")]
37         public ActionResult Hakkimizda()
38         {
39             using (TimeZonaEntities db = new TimeZonaEntities())
40             {
41                 var model = db.ABOUTUS.ToList();
42                 return View(model);
43             }
44         }
45     }
46 }

```

DefaultController kodları


```

public ActionResult Kaydet(PRODUCTSDETAILS gelenUrun)
{
    using (TimeZonaEntities db = new TimeZonaEntities())
    {
        if (!ModelState.IsValid) //Formun doğru dolduruldu mu?
        {
            return View("UrunDetayForm", gelenUrun);
        }

        if (gelenUrun.ID == 0) //Yeni ürün kaydı
        {
            if (gelenUrun.fotoFile == null)
            {
                ViewBag.HataFoto = "Lütfen Resim Yükleyin";
                return View("UrunDetayForm", gelenUrun);
            }
            string fotoAdi = Seo.DosyaAdiDuzenle(gelenUrun.fotoFile.FileName);

            gelenUrun.PHOTO = fotoAdi;
            db.PRODUCTSDETAILS.Add(gelenUrun);
            gelenUrun.fotoFile.SaveAs(Path.Combine(Server.MapPath("~/Content/img"), Path.GetFileName(fotoAdi)));
            TempData["urunEkle"] = "ekle";
        }
        else //Güncelleme
        {
            var GuncellenecekVeri = db.PRODUCTSDETAILS.Find(gelenUrun.ID);
            if (!ModelState.IsValid)
            {
                return View("UrunDetayForm", gelenUrun);
            }

            if (gelenUrun.fotoFile != null)
            {
                string fotoAdi = Seo.DosyaAdiDuzenle(gelenUrun.fotoFile.FileName);
                gelenUrun.PHOTO = fotoAdi;
                //Kaydetme
                gelenUrun.fotoFile.SaveAs(Path.Combine(Server.MapPath("~/Content/img"), Path.GetFileName(fotoAdi)));

                //güncelleme
                db.Entry(GuncellenecekVeri).CurrentValues.SetValues(gelenUrun);
                TempData["urun"] = "güncelleme";
            }

            db.SaveChanges();
            return RedirectToAction("/index");
        }
    }
}

public ActionResult Sil(int id)
{
    using (TimeZonaEntities db = new TimeZonaEntities())
    {
        var silinecekVeri = db.PRODUCTSDETAILS.Find(id);

        if (silinecekVeri == null)
        {
            return HttpNotFound();
        }

        db.PRODUCTSDETAILS.Remove(silinecekVeri);
        db.SaveChanges();

        TempData["urun"] = "silindi";
        return RedirectToAction("/index", "Urunler");
    }
}

```

UrunlerDetayController kodları-Ürün kaydı,güncelleme,silme işlemleri

Web sayfalarının html kodları ve tüm dosyalar githubda yer almaktadır.

4. API

Sitede oluşturulan blog, ürün detayları gibi bilgilerin api vasıtasıyla gönderilmesi ve gösterilmesine yönelik kodlamalar yapılmıştır. Bu doğrultuda en çok tercih edilen saatlerin kaç kere tercih edildiği gibi bilgileri içeren bir api oluşturmak için “controller” kullanılarak “ApiController” oluşturulmuştur.