İSTANBUL TOPKAPI

ÜNİVERSİTESİ

MVC İLE DİNAMİK WEB SİTESİ

DR. MEHMET AKİF ÇİFÇİ

ZEHRA DÖNMEZ - 20040301033 - [zehradonmez@stu.topkapi.edu.tr](mailto:zehradonmez@stu.topkapi.edu.tr)

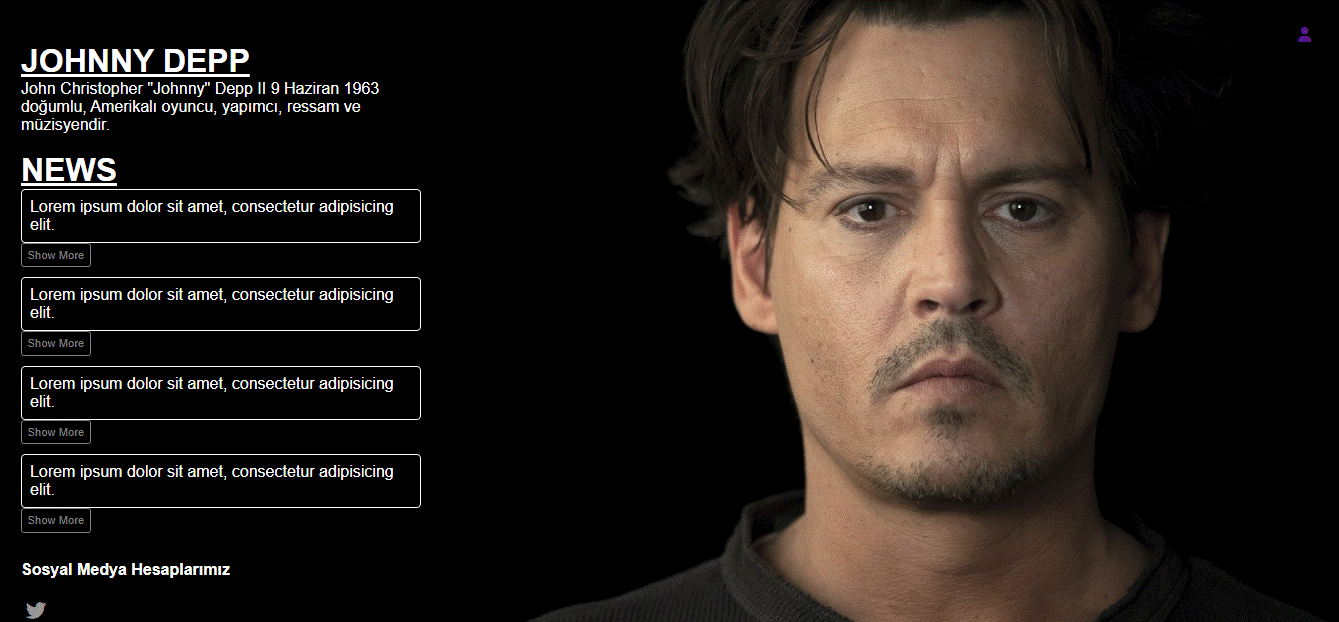
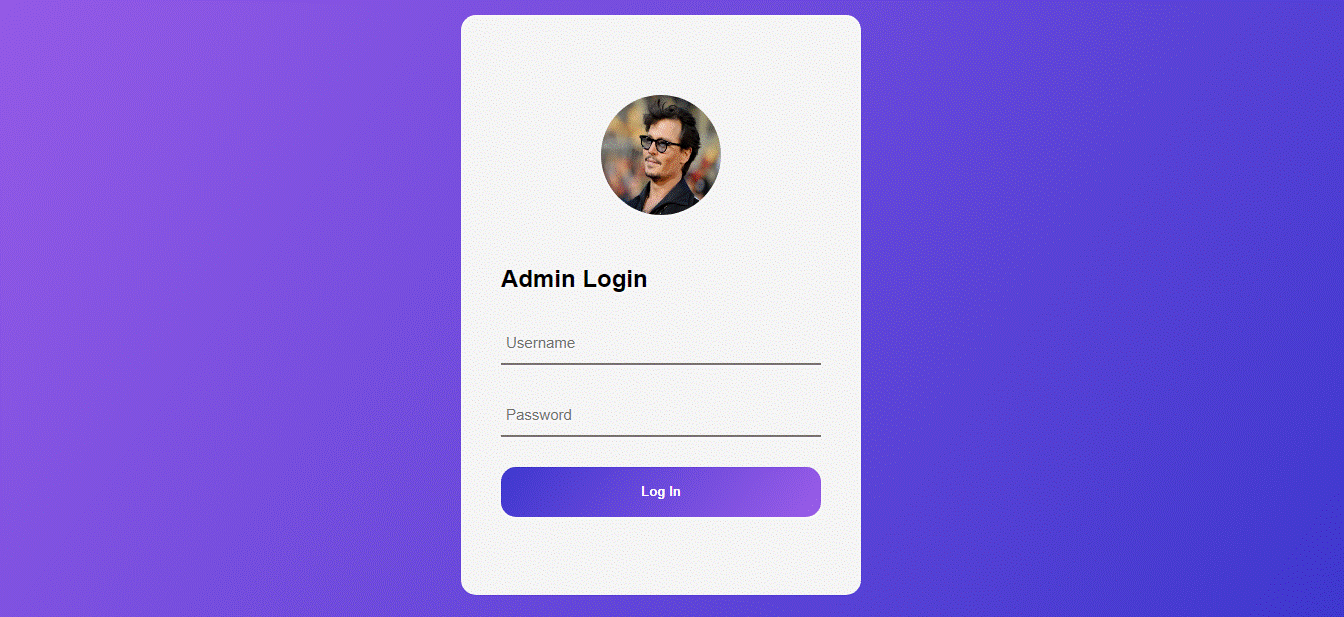
**Index Sayfalarının Temalarının Oluşturulması**

Web-Admin-Login ve Web-Tufod-Index adlı iki dosya oluşturduk. Css ve html sayfalarını da oluşturarak temamızı oluşturduk. Image dosyaları ile fotoğraflarımızı ekledik.

Ana Sayfa Indexinde classlar oluşturduk ve genel şemayı kurduk.

Küçük bir tanıtım alanı, haberler alanı, sosyal medya alanı olacak şekilde sitemizin sol tarafını detaylandırdık. Sağ tarafta sadece background image gözükecek şekilde ayarlamalarımızı yaptık.

Login Paneli Indexinde de sayfanın ortasında login alanı kalacak şekilde ayarlamamızı yaptık.



Temalarımız bu şekilde gözükmekteydi.

**MVC Dosyalarının Oluşturulması**

HomePage controllerimizi ve Index sayfalarımızı oluşturduk. Birbirleri arasında bağlantımızı kurup Index sayfamıza önceden hazırladığımız Ana Sayfa Index’ini ekledik.

Modellerde bir Class oluşturduk ve Web Config ayarlarını sitemize uygun güncelledik. Migrations işlemlerini de bitirdikten sonra HomePage Controller yardımı ile Ana Sayfamızı Listeledik.

Migrations işlemlerinde hazırladığımız veritabanı tabloları içerisinden veri alacak şekilde foreach döngüsü kullandık.

Veritabanında ikon bağlantılarını da kurduk.

**HOMEPAGE CONTROLLER**

using FinalTUFOD.Models.Classes;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace FinalTUFOD.Controllers

{

public class HomePageController : Controller

{

// GET: HomePage

Context c = new Context();

public ActionResult Index()

{

var variable = c.HomePages.ToList();

return View(variable);

}

public PartialViewResult icons()

{

var variable = c.icons.ToList();

return PartialView(variable);

}

}

}

**Admin Panelinin Kurulması**

\_AdminLayout içerisinde bootstrap ile bir tema düzenledik. Admin Controller ile bağlantısını oluşturduktan sonra belirttiğimiz linklerde yapılacak işlemleri oturttuk.

Ana sayfa güncelleme ekranında update işlemlerini tamamladık.

Ikon sayfasında ikon ekleme, güncelleme ve silme işlemlerini de Admin Controllerde oluşturduğumuz fonksiyonlar yardımıyla sağladık. GoToHomePage, GoToIcon, iconLİst ve newIcon bu fonksiyonlardır.

Hazırladığımız login paneli ile admin paneli için birbirine bağlantı kurduk. Veritabanında önceden hazırlanılmış admin tablosunda bir kullanıcı oluşturduk. Login controller ile giriş ve çıkış işlemlerini güvenlikle sağladık.

Web configte açılan form etiketi içerisinde kimlik doğrulama işlemlerini de sağladıktan sonra projemiz bitti.

**ADMIN CONTROLLER**

using FinalTUFOD.Models.Classes;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace FinalTUFOD.Controllers

{

public class AdminController : Controller

{

// GET: Admin

Context c = new Context();

[Authorize]

public ActionResult Index()

{

var variable = c.HomePages.ToList();

return View(variable);

}

public ActionResult GoToHomePage(int id)

{

var gth = c.HomePages.Find(id);

return View("GoToHomePage", gth);

}

public ActionResult UpdateHomePage(HomePage x)

{

var hp = c.HomePages.Find(x.id);

hp.name = x.name;

hp.content = x.content;

hp.newshead = x.newshead;

hp.contant = x.contant;

hp.news1head = x.news1head;

hp.news2head = x.news2head;

hp.news3head = x.news3head;

hp.news4head = x.news4head;

hp.news1 = x.news1;

hp.news2 = x.news2;

hp.news3 = x.news3;

hp.news4 = x.news4;

c.SaveChanges();

return RedirectToAction("Index");

}

public ActionResult iconList()

{

var variable = c.icons.ToList();

return View(variable);

}

[HttpGet]

public ActionResult newIcon()

{

return View();

}

[HttpPost]

public ActionResult newIcon(Icons p)

{

c.icons.Add(p);

c.SaveChanges();

return RedirectToAction("iconList");

}

public ActionResult GoToIcon(int id)

{

var gti = c.icons.Find(id);

return View("GoToIcon", gti);

}

public ActionResult UpdateIcon(Icons x)

{

var ic = c.icons.Find(x.id);

ic.icon = x.icon;

ic.link = x.link;

c.SaveChanges();

return RedirectToAction("iconList");

}

public ActionResult DeleteIcon(int id)

{

var di = c.icons.Find(id);

c.icons.Remove(di);

c.SaveChanges();

return RedirectToAction("iconList");

}

}

}

**LOGIN CONTROLLER**

using FinalTUFOD.Models.Classes;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using System.Web.Security;

namespace FinalTUFOD.Controllers

{

public class LoginController : Controller

{

// GET: Login

Context c = new Context();

public ActionResult Index()

{

return View();

}

[HttpPost]

public ActionResult Index(AdminPage ad)

{

var info = c.Admins.FirstOrDefault(x => x.username == ad.username && x.password == ad.password);

if(info != null)

{

FormsAuthentication.SetAuthCookie(info.username, false);

Session["username"] = info.username.ToString();

return RedirectToAction("Index", "Admin");

}

else

{

return View();

}

}

public ActionResult LogOut()

{

FormsAuthentication.SignOut();

return RedirectToAction("Index", "Login");

}

}

}

Web config ayarları ile sitemize url ile sızılmasının önüne geçtik.