

WEB PROGRAMLAMA PROJE ÖDEVİ RAPORU

Öğrencinin;

Adı: Yusuf

Soyadı: Sönmez

Numarası: B181210071

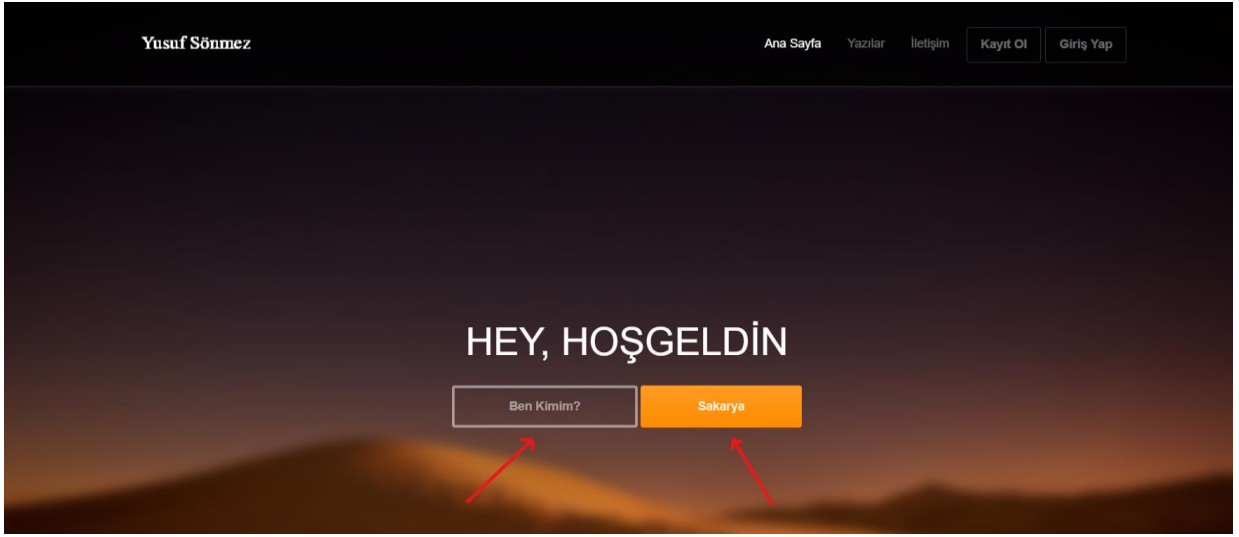
Projenin GitHub linki:

https://github.com/YusufSonmezz/web_programming_project

Proje Tanıtımı

Proje, kendi yazılarımı yayınladığım, insanlarla etkileşim halinde olduğum ve kendimi anlattığım bir web sitesini kapsamaktadır.

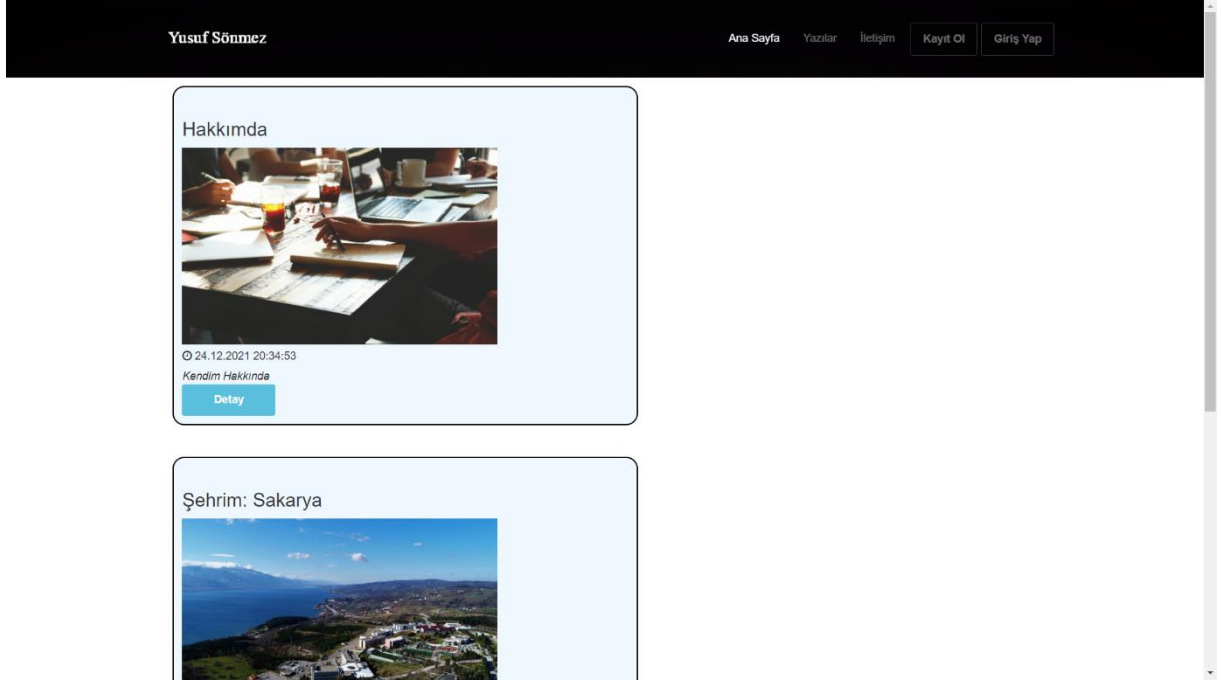
Projeyi tasarlarken kullanıcıyla interaktif bir biçimde etkileşebileceğimiz bir ortamın olmasını göz önünde bulundurdum. Kullanıcının istediği zaman bana mesaj gönderebileceği, yazılarıma yorum yapabileceği ve sosyal medya hesaplarıma kolayca ulaşabileceği tasarımları canlandırmaya gayret ettim.



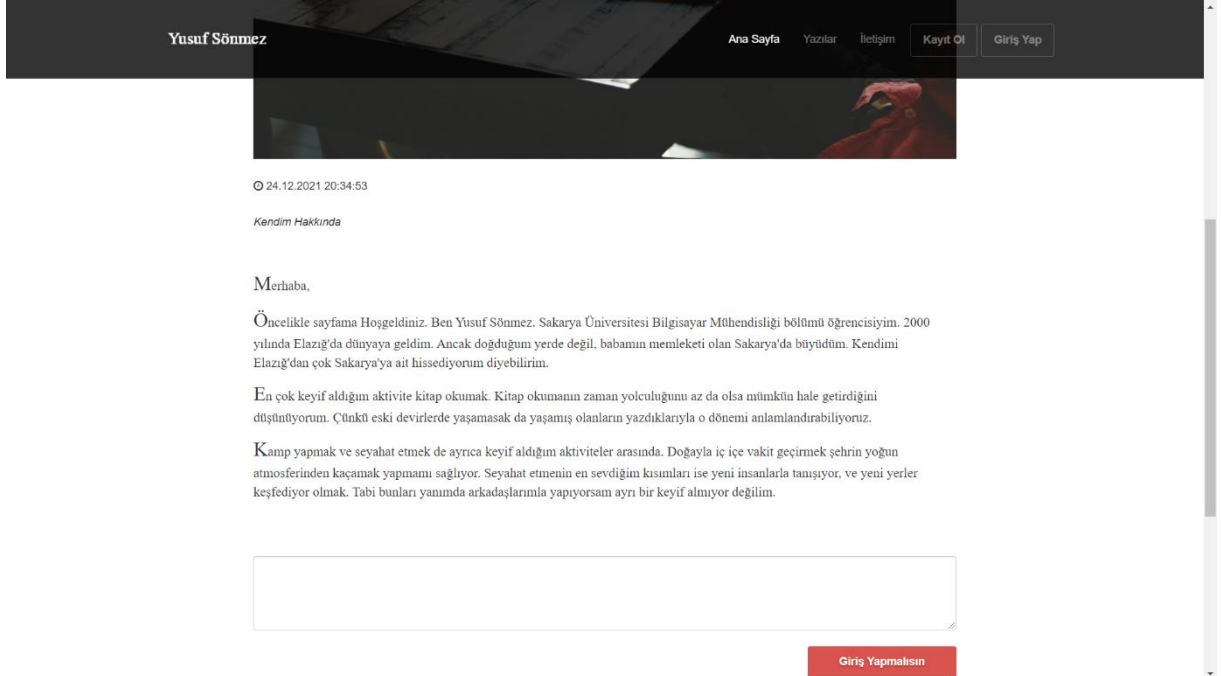
Yukarıdaki resimde kırmızı renklerle gösterilen iki adet buton, yeni gelen ziyaretçiye bilgilendirmek istediğim iki konu -kendim ve yaşadığım şehir- hakkında yazdığım yazıların bulunduğu linklere onları götürecek şekilde tasarlandı.

Ziyaretçinin ilk karşılaşacağı sayfa olan Ana Sayfada neden bu siteyi kurduğuma ve kendim ile alakalı konulara dair küçük ipuçları edinebilecek.

Yazılar kısmında ise yazdığım tüm yazılara erişim mümkün. Oradan yazdığım mesleki, kişisel ya da hobilerimle alakalı yazılara dinamik bir şekilde ulaşabilecek. Yazıların, yazılma tarihi ve saati, yazıya ait resim, başlık ve bir açıklama ile daha anlaşılır olması amaçlandı.



Detaya tıklayarak yazının tamamına erişilebilir ve yazıya yorum yapılabilir. Ancak yorum yapmak için öncelikle siteye kaydolma şartı aranmaktadır. Eğer bir kullanıcı kaydolmuş ise dilediği yorumu yapabilmektedir.



Yukarıda da görüldüğü gibi giriş yapmadığımız için yorum kısmındaki buton kırmızı ve pasif. Eğer giriş yapılmış olsaydı bu buton turuncu olacaktı ve üzerinde Yorum Yap yazacaktı.

Yorumlar dinamik bir şekilde ekrana yansıtılmaktadır. Yorum Yap butonuna tıklayan bir kullanıcının yorumu öncelikle veri tabanına kaydolur, daha sonra veri tabanından çekilerek ilgili yazının altına tarih ve kullanıcı adı ile yazılır.

Düzenleyen Kişi: b181210071@sakarya.edu.tr
Düzenlendiği Tarih: 25.12.2021 02:03:49

Örnek Yorumdur!!!

Yorum Yap

İletişim kısmı ise gerekli bilgileri ve mesajın içeriğini kullanıcıdan alarak admin panelinde gözükecek şekilde veri tabanına kaydetmektedir.

```
namespace WebProgProje.Models
{
    public class Comment
    {
        [Key]
        public int CommentId { get; set; }

        [Required(ErrorMessage = "İçerik boş bırakılmamalı")]
        [Display(Name = "İçerik")]
        public string Content { get; set; }

        [Required]
        public DateTime DateOfCommenting { get; set; }

        public string UserName { get; set; }

        public int ArticleId { get; set; }

        public Article article { get; set; }
    }
}
```

```

namespace WebProgProje.Models
{
    public class Contact
    {
        [Key]
        public int ContactId { get; set; }

        [Required(ErrorMessage = "Adını yazarsan sana hitap etmem kolaylaşır")]
        [Display(Name = "İsim")]
        public string Name { get; set; }

        [Required(ErrorMessage = "Sana e-mailin aracılığıyla ulaşabilirim")]
        [EmailAddress]
        [Display(Name = "E-Mail")]
        public string Email { get; set; }

        [Phone]
        [Display(Name = "Telefon")]
        public string Phone { get; set; }

        [Required(ErrorMessage = "Boş içerik göndermemelisin")]
        [Display(Name = "İçerik")]
        public string Content { get; set; }

        [Display(Name = "Okundu")]
        public bool isRead { get; set; }

        [Required]
        [Display(Name = "İleti Tarihi")]
        public DateTime DateOfComment { get; set; }
    }
}

```

```

namespace WebProgProje.Models
{
    public class Article
    {
        [Key]
        public int ArticleId { get; set; }

        [Required(ErrorMessage = "Başlık boş bırakılmamalı")]
        [Display(Name = "Başlık")]
        public string Title { get; set; }

        [Required(ErrorMessage = "Açıklama boş bırakılmamalı")]
        [Display(Name = "Açıklama")]
        public string Description { get; set; }

        [Display(Name = "Resim")]
        public string Image { get; set; }

        [Required(ErrorMessage = "İçerik boş bırakılmamalı")]
        [Display(Name = "İçerik")]
        public string Content { get; set; }

        [Required]
        public DateTime DateOfAdding { get; set; }

        public ICollection<Comment> Comments { get; set; }
    }
}

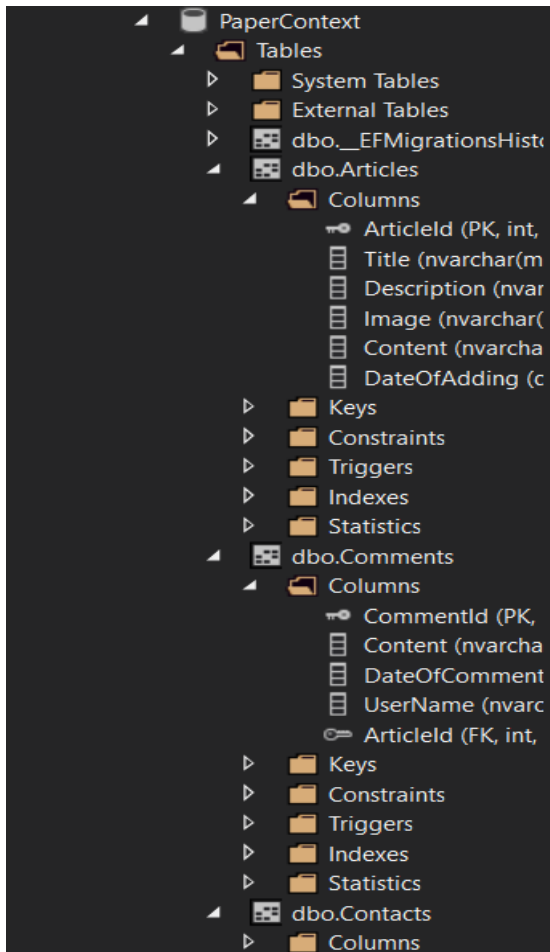
```

Yukarıdaki 3 adet sınıf sırasıyla: Yorum => Yazı => İletişim sınıflarıdır. Bu sınıflar alınan bilgileri depolamak için tasarlanmışlardır. Bu modelleri kullanarak DbContext sınıfından da kalıtım alarak veri tabanında ilişkisel bir model kurulmuştur.

```
namespace WebProgProje.Models
{
    public class PaperContext : DbContext
    {
        public DbSet<Article> Articles { get; set; }
        public DbSet<Comment> Comments { get; set; }
        public DbSet<Contact> Contacts { get; set; }

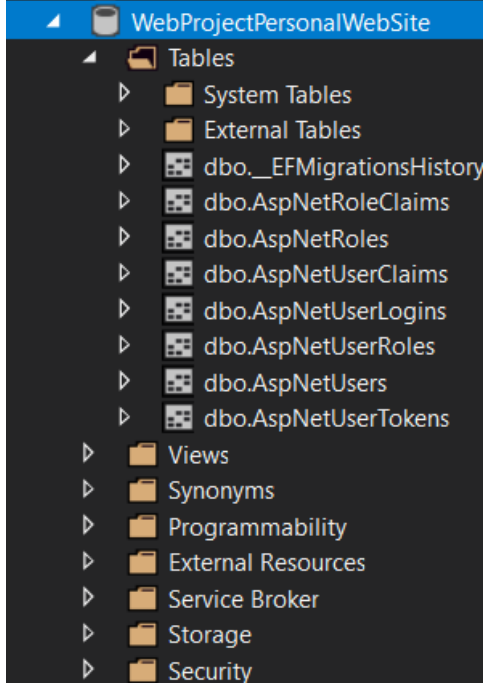
        public PaperContext(DbContextOptions<PaperContext> options) : base(options)
        {
        }
    }
}
```

PaperContext ismi altında bu sınıflar toplanmış ve bir veri tabanı oluşturulmuştur.



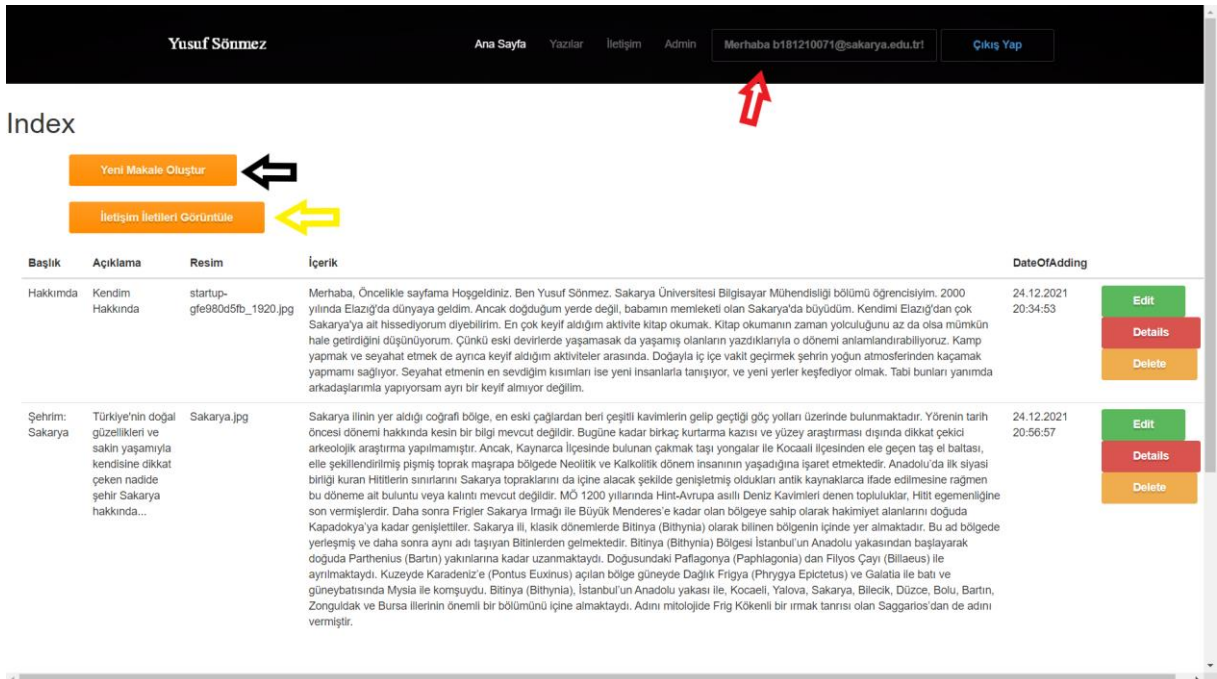
Yan tarafta görüldüğü gibi PaperContext sınıfına DbContext sınıfından kalıtım aldık ve migration ile de CodeFirst yaklaşımı ile veri tabanımızı oluşturduk. Comments altındaki ArticleId ye bakacak olursanız görebilirsiniz ki sistem otomatik olarak ArticleId yi algılamış ve ona göre bir foreign key tanımlamıştır.

Ayrıca projeyi başlatırken seçeneklerden seçtiğimiz Windows Authentication ile de ayrı bir Context oluşturuluyor. Bu Contextin içerisinde de kullanıcılar ile alakalı bilgiler, kullanıcıların rolleri vs. kategorik bilgiler tutuluyor ve kontrolleri sağlanıyor.



Yan tarafta gözüken veri tabanı sistemin kaydolma ve giriş yapma eylemlerini kontrol etmektedir. Kullanıcıların kaydolurken girdikleri bilgiler, kullanıcıların atandıkları roller -admin, user vs.- bu gibi bilgileri tutar ve kontrol eder.

Admin Panelinde ise gelen iletileri görebilme, yazıları silme, düzenleme, detaylarına göz atma ve yeni yazı oluşturma gibi seçenekler mevcut.



Bir üstteki resimde **kırmızı ok** ile gösterilen butonda Merhaba <kullanıcı ismi> yazıyor. Bu bizim giriş yaptığımızı gösteriyor. Ayrıca eklediğim ilk resimdeki menüde Admin seçeneği yokken şu anda Admin seçeneği gelmiş gözüküyor.

Siyah oka tıkladığımızda da diğer bir seçenek olan yeni yazı oluşturabildiğimiz sayfaya gidiyoruz.

Sarı oka tıkladığımızda ise bize iletilen mesajlardan okunmamış olanları görebilmekteyiz.

İletiler

İsim	E-Mail	Telefon	İçerik	Okundu	İleti Tarihi	
Yusuf Sönmez	yusufsn54@gmail.com	0(554) 660-3782	Yusuf selamlar Sana söylemek istediğim: Nasılsın?	<input type="checkbox"/>	22.12.2021 01:20:00	Okundu Olarak İşaretle
Kayra Çelik	kayracelik@hotmail.com	5557865982	Merhaba Yusuf Bey, sitenizi ziyaret ederken...	<input type="checkbox"/>	25.12.2021 02:20:40	Okundu Olarak İşaretle

Okundu olarak işaretli butonuna tıklarsak, bir daha o mesaj karşımıza çıkmayacaktır. Böylelikle gereksiz bir yoğunluk ileti kısmında oluşmamış oluyor.

Tüm bu okların altında var olan yazıların listelendiğini görüyoruz. Yazılarla alakalı bilgilerin hepsi bu listede mevcut. Ayrıca yanında yer alan Edit, Details, Delete butonları da yazılar üzerinde değişiklikler yapmamızı kolaylaştırıyor.