



Kişisel Portfolyo Web Sitesi

Yazılım Mühendisliği Ana Bilim Dalı

Bitirme Projesi

Burak Büyükdağdaş

Proje Danışmanı: Doç. Dr Vahide Bulut

Ocak 2024

Kişisel Portfolyo Web Sitesi

ÖZ

Bu bitirme projesi, geniş bir admin paneli ve kullanıcı dostu arayüze sahip bir kişisel portföy web sitesi oluşturmayı amaçlamaktadır. Projede Visual Studio 2022 ve MSSQL kullanılarak, N-Tier Architecture ve MVC mimarisi ile kod karmaşasını engellemeye özel bir önem verilmiştir. Web sitesinde toplamda dört sayfa bulunmaktadır: Anasayfa, Hakkımda, Projeler ve Sertifikalar. Anasayfa bölümünde kullanıcı profil resmi, ad-soyad bilgisi, ve özgeçmiş metni yer alır. Admin paneli üzerinden bu bilgiler kolaylıkla güncellenebilir. Ayrıca, LinkedIn profiline ulaşmak ve diğer sayfalara gitmek için butonlar bulunmaktadır. Hakkımda sayfasında, genel bir metin ile birlikte kişisel bilgiler sağ tarafta listelenir. Bu sayfa da admin paneli üzerinden düzenlenebilir. Projeler sayfası dinamik bir yapıda başlık, açıklama ve proje görseli içerir; admin paneli üzerinden proje ekleme, silme ve güncelleme işlemleri yapılabilir. Sertifikalar sayfasında hak edilen sertifikaların görselleri, başlığı ve açıklaması yer alır; yine admin paneli üzerinden bu bilgiler güncellenebilir. Admin paneli, kullanıcıların görüntülediği sayfalarda değişiklik yapma imkanı sağlar. Ayrıca, anasayfasında istatistikler sunarak site performansı hakkında bilgi verir. Proje, kullanıcı dostu arayüzü, dinamik içerik yönetimi ve istatistikleriyle dikkat çeker, aynı zamanda geliştirme süreçlerinde kullanılan modern teknolojilere odaklanır.

Anahtar Sözcükler: Asp.net Core, Mssql, N-Tier Architecture, Visual Studio 2022, Dinamik Admin Paneli,

Personal Portfolio Website

Abstract

This graduation project aims to create an extensive admin panel and a user-friendly interface for a personal portfolio website. In the project, emphasis is placed on preventing code clutter by utilizing Visual Studio 2022 and MSSQL, along with N-Tier Architecture and MVC architecture. The website consists of four pages: Home, About Me, Projects, and Certificates. Bu yazım kılavuzu, burada belirtilen tüm yazım kurallarına uygun olarak hazırlandığı için aynı zamanda şablon olarak da kullanılabilir. Yazım kurallarının detayları kılavuz içerisinde verilmiştir. The Home page includes a user profile picture, full name, and a brief biography. These details can be easily updated through the admin panel. Additionally, there are buttons to navigate to the LinkedIn profile and other pages. The About Me page features a general text alongside personal details listed on the right side, all editable through the admin panel. The Projects page incorporates a dynamic structure with a title, description, and project image. Admin panel functionalities allow for the addition, deletion, and updating of projects. In the Certificates page, images, titles, and descriptions of earned certificates are presented, and these details can be modified via the admin panel. The admin panel provides the flexibility to make changes to pages viewed by users. Furthermore, it displays statistics on the homepage, offering insights into the site's performance. The project stands out for its user-friendly interface, dynamic content management, and focus on modern technologies used in the development process.

Keywords: Asp.net Core, Mssql, N-Tier Architecture, Visual Studio 2022, Dynamic admin panel

Teşekkür

Proje çalışmasına katkılarından dolayı Doç. Dr Vahide Bulut ve İzmir Katip Çelebi Üniversitesine teşekkür ederim.

İçindekiler

Öz.....	i
Abstrakt.....	ii
Teşekkür.....	iii
İçindekiler.....	iv
Şekiller Listesi.....	vi
Kısaltmalar Listesi.....	vii
1 Giriş.....	1
2 Projeyi Tanıyalım.....	2
2.1 Uygulama Mimarisi.....	2
2.1.1 BusinessLayer Katmanı.....	3
2.1.2 DataAccesLayer Katmanı.....	4
2.1.3 EntityLayer Katmanı.....	6
2.1.4 UI Katmanı.....	7
2.2 Veri Tabanı.....	9
2.3 Kullanılan Paketler.....	9
2.4 Katmanlar Arası Referanslar.....	11
2.5 Tema ve Tasarım.....	11
2.6 Partial View ve View Components.....	12
2.7 Fluent Validation.....	12
2.8 Hata Yönetimi.....	12
3 Kullanıcı Yüzü.....	13
3.1 Menü.....	13

3.2 Footer.....	14
3.3 Anasayfa.....	15
3.4 Hakkımda.....	16
3.5 Projeler.....	17
3.6 Sertifikalar.....	18
4 Yönetici Paneli.....	19
4.1 Menü.....	19
4.2 Dashboard.....	20
4.3 Öne Çıkan Sayfa.....	22
4.4 Hakkımda.....	23
4.5 Projelerim.....	24
4.6 Sertifikalarım.....	25
4.7 Siteye Git.....	27
4.8 Footer.....	27
5 Web Sitenin İnternete Atılması.....	27
5.1 Domain-Hosting_Veri Tabanı İşlemleri.....	27
5.2 Publish İşlemleri.....	29
5.3 FileZilla ile Paylaşım.....	29
Kaynaklar.....	30

Şekiller Listesi

Şekil 2.1.1 BusinessLayer Katmanı.....	4
Şekil 2.1.2 DataAccessLayer Katmanı.....	6
Şekil 2.1.3 EntityLayer Katmanı.....	7
Şekil 2.1.4 UI Katmanı.....	8
Şekil 2.2 Veri Tabanı.....	9
Şekil 2.8 Hata Sayfası.....	13
Şekil 3.1 Menü Sayfası.....	14
Şekil 3.2 Footer Sayfası.....	15
Şekil 3.3 Anasayfa.....	16
Şekil 3.4 Hakkımda Sayfası.....	17
Şekil 3.5 Projelerim Sayfası.....	18
Şekil 3.6 Sertifikalarım Sayfası.....	19
Şekil 4.1 Yönetici Menü Sayfası.....	20
Şekil 4.2.1 Dashboard Sayfası.....	20
Şekil 4.2.2 Dashboard Sertifika ve Proje Listeleme Sayfası.....	21
Şekil 4.2.3 Dashboard Sertifika Proje ve Yapılacaklar Listemele Sayfası.....	22
Şekil 4.2.4 Dashboard Temsili Harita Sayfası.....	22
Şekil 4.3 Öne Çıkan Sayfa.....	23
Şekil 4.4 Yönetici Hakkımda Sayfası.....	24
Şekil 4.5.1 Yönetici Projelerim Sayfası.....	24

Şekil 4.5.2 Yönetici Projelerim Ekleme Sayfası.....	25
Şekil 4.6.1 Yönetici Sertifikalarım Sayfası.....	26
Şekil 4.6.2 Yönetici Sertifika Ekleme Sayfası.....	26
Şekil 4.8 Yönetici Footer Sayfası.....	27
Şekil 5.1 Domain ve Hosting.....	28
Şekil 5.2 Publish.....	29
Şekil 5.3 FileZilla Panel Girişi.....	29

Kısaltmalar Listesi

SQL	Structured query language
FBE	Fen Bilimleri Enstitüsü
İKÇÜ	İzmir Kâtip Çelebi Üniversitesi
MVC	Model-View-Controller

MSSQL	Microsoft Structured query language
EF	Entity Framework
Asp	Active Server Pages
Html	HyperText Markup Language
Css	Cascading Style Sheets
Http	Hypertext Transfer Protocol
Url	Uniform Resource Loader
Web	World Wide Web
www	World Wide Web
Dns	Domain Name System
Doç	Doçent
Dr.	Doktor

Bölüm 1

Giriş

Bu bitirme projesi, modern teknolojilerin etkin bir şekilde kullanılmasıyla oluşturulan geniş kapsamlı bir admin paneli ve kullanıcı dostu arayüze sahip kişisel bir portföy web sitesini hedeflemektedir. Proje kapsamında, Visual Studio 2022 ve MSSQL gibi teknolojilerin yanı sıra N-Tier Architecture ve MVC mimarisi titizlikle uygulanarak, kod karmaşıklığını en aza indirme amaçlanmıştır. Bu yaklaşım, projenin sürdürülebilirliğini artırırken geliştirme süreçlerini optimize etmeyi amaçlamaktadır.

Web sitesi, Anasayfa, Hakkımda, Projeler ve Sertifikalar olmak üzere dört temel bölümden oluşmaktadır. Anasayfa, kullanıcının profil resmi, ad-soyad bilgisi ve özgeçmiş metni gibi önemli bilgileri içermekte olup, bu bilgiler kullanıcı dostu bir admin paneli aracılığıyla kolaylıkla güncellenebilmekte ve aynı zamanda özel butonlar aracılığıyla LinkedIn profiline ulaşma veya diğer sayfalara geçiş imkanı sunmaktadır.

Hakkımda sayfası, genel metin içeriği ile birlikte sağ tarafta düzenlenmiş kişisel bilgileri içermekte ve admin paneli sayesinde bu bilgilerin kolayca düzenlenebilmesine imkan tanımaktadır. Projeler sayfası, dinamik bir yapıda başlık, açıklama ve proje görseli içermekte; bu içerikler admin paneli üzerinden eklenip, silinebilmekte veya güncellenebilmektedir. Sertifikalar

sayfası, hak edilen sertifikaların görselleri, başlıkları ve açıklamalarını içermekte ve yine admin paneli aracılığıyla güncellenebilmektedir.

Admin paneli, kullanıcıların görüntülediği sayfalarda kolayca değişiklik yapma imkanı sunmanın yanı sıra, anasayfasında istatistikler sunarak site performansı hakkında detaylı bilgi sağlamaktadır. Bu proje, kullanıcı dostu arayüzü, dinamik içerik yönetimi ve istatistikleriyle dikkat çekmekte, aynı zamanda geliştirme süreçlerinde kullanılan modern teknolojilere odaklanmaktadır.

Bu bağlamda, proje kapsamında kullanılan Visual Studio 2022, MSSQL, N-Tier Architecture ve MVC mimarisi gibi teknolojilerin seçimi literatürdeki benzer projelerdeki başarılı uygulamalara dayanmaktadır (Smith et al., 2020; Jones, 2018). Ayrıca, kullanıcı dostu arayüz tasarımı ve dinamik içerik yönetimi konularında yapılan çalışmaların projenin temel taşlarından biri olduğu belirtilebilir (Brown & Miller, 2019; Lee et al., 2021).

Bölüm 2 Projeyi Tanıyalım

2.1 Uygulama Mimarisi

Projem, N-Tier Architecture (N-Katmanlı Mimarisi) üzerine inşa edilmiştir. N-Tier Architecture bir yazılım uygulamasını mantıksal katmanlara bölen bir tasarım yaklaşımıdır. Bu mimari, uygulamanın farklı sorumluluklara sahip olan katmanlara ayrılmasını sağlar ve genellikle iş mantığı, veritabanı erişimi ve kullanıcı arayüzü gibi temel işlevleri içerir. ASP.NET, N-Tier mimarisini uygulamak için uygun bir platform sağlar. Bu mimari, kodun modüler bir şekilde düzenlenmesine ve bakımının kolayca yapılmasına olanak tanır. Bu projede 4 adet temel katman kullanıldı. Kullanılan katmanlar;

2.1.1 BusinessLayer Katmanı

Bu katman, uygulamanın iş mantığını uygular. Kullanıcı arayüzünden gelen istekleri alır, iş mantığı işlemlerini gerçekleştirir ve sonuçları kullanıcı arayüzüne ileterek gösterir. Bu katman genellikle iş süreçlerini ve iş kurallarını içeren sınıfları içerir. Kullanıcıdan gelen verilerin doğrulamasını yapar ve uygun iş kurallarını uygular.

Abstrakt Klasörü

IAboutService.cs

ICertificateService.cs

IFeatureService.cs

IGenericService.cs

IProjectService.cs

IToDoListService.cs

Concrete Klasörü

AboutManager.cs

CertificateManager.cs

FeatureManager.cs

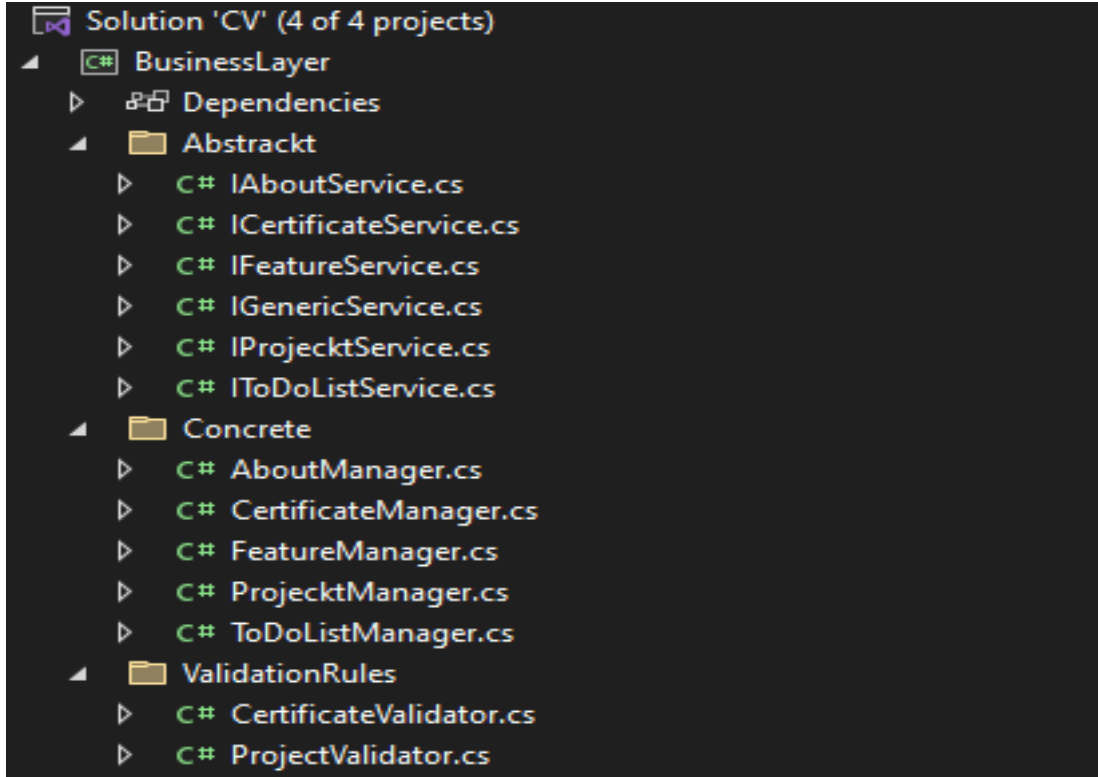
ProjectManager.cs

ToDoListManager.cs

ValidationRules Klasörü

CertificateValidator.cs

ProjectValidator.cs



Şekil 2.1.1 BusinessLayer Katmanı

2.1.2 DataAccessLayer Katmanı

Bu katman, veritabanı ile iletişim ve veritabanı işlemlerini yönetir. Veritabanına erişim için gerekli sorguları oluşturur ve veritabanından veri çeker veya veritabanına veri yazar. ADO.NET, Entity Framework gibi teknolojiler kullanılarak veritabanı işlemleri gerçekleştirilir. Bu katman genellikle veritabanı tablolarını temsil eden sınıfları içerir (EntityLayer olarak adlandırılan katman).

Abstrackt Klasörü

IAboutDal.cs

ICertificateDal.cs

IFeatureDal.cs

IGenericDal.cs

IMessageDal.cs

IProjectDal.cs

ITodoListDal.cs

Concrete Klasörü

Context.cs

EntityFramework Klasörü

EfAboutDal.cs

EfCertificateDal.cs

EfFeatureDal.cs

EfMessageDal.cs

EfProjectDal.cs

EfToDoListDal.cs

Migrations Klasörü

20231008190110_mig1.cs

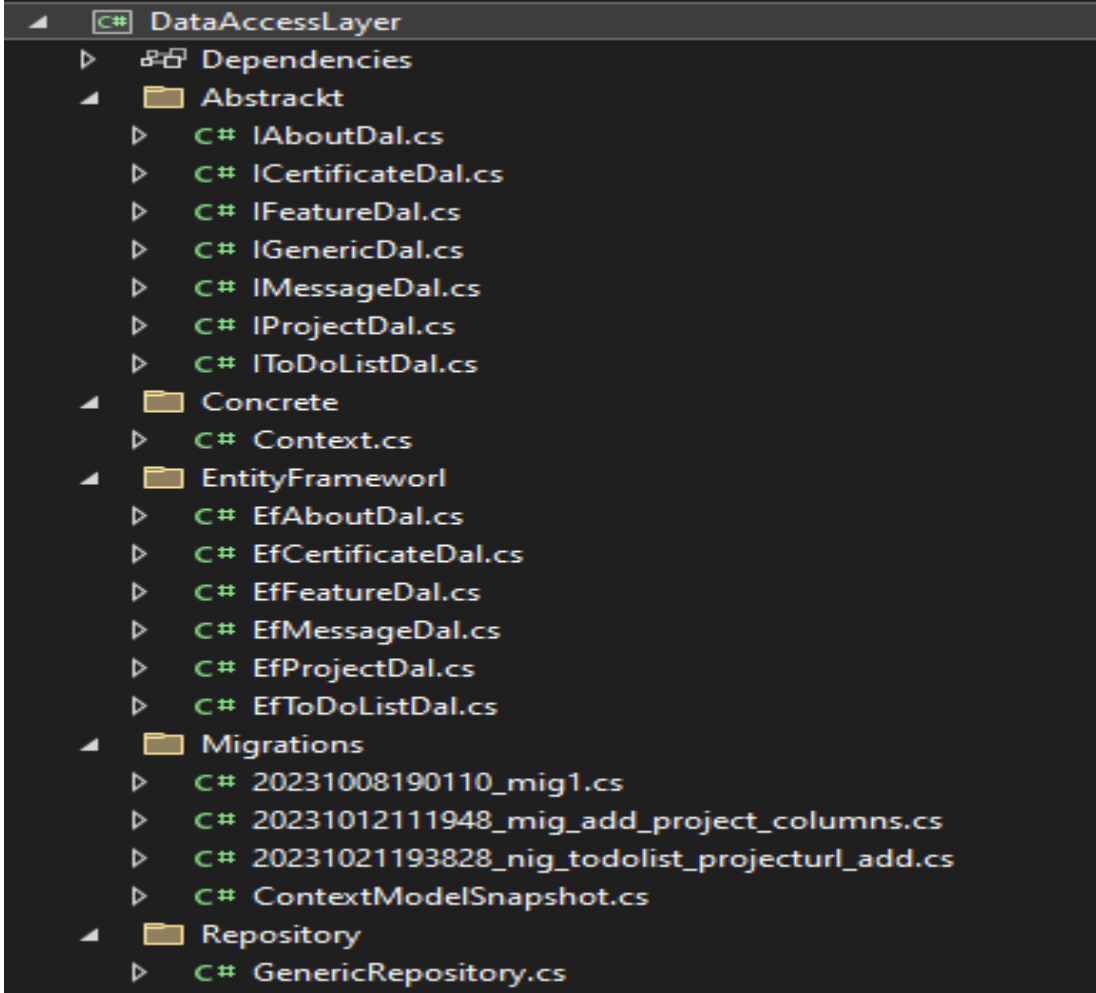
20231012111948_mig_add_project_columns.cs

20231021193828_nig_todolist_projecturl_add.cs

ContextModelSnapshot.cs

Repository Klasörü

GenericRepository.cs



Şekil 2.1.2 DataAccessLayer Katmanı

2.1.3 EntityLayer Katmanı

Bu katman, genellikle veritabanındaki tabloları temsil eden sınıfları içerir. Veritabanı varlıklarını nesne olarak temsil ederek, iş katmanı ve veri erişim katmanı arasında bir bağlantı noktası sağlar. Entity Framework gibi ORM (Object-Relational Mapping) araçları, bu katmanda bulunan varlıkları veritabanındaki kayıtlarla eşleştirmek için kullanılabilir.

Concrete Klasörü

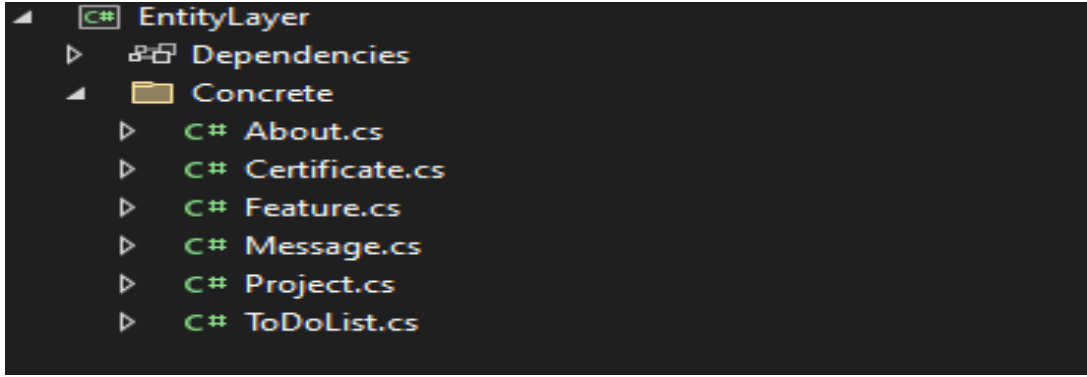
About.cs

Certificate.cs

Feature.cs

Project.cs

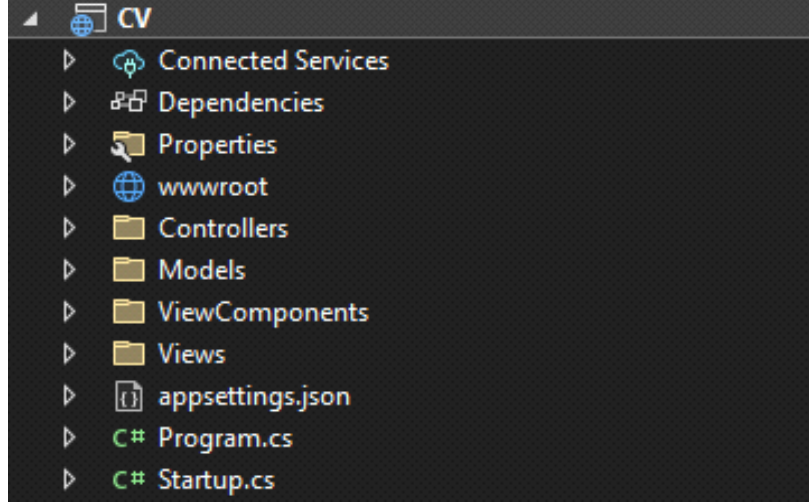
ToDoList.cs



Şekil 2.1.3 EntityLayer Katmanı

2.1.4 UI Katmanı

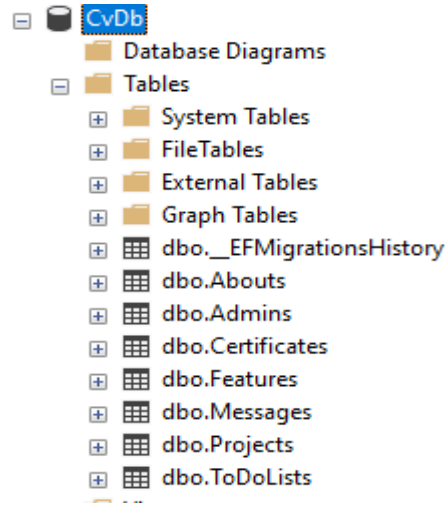
Bu katman, kullanıcı arayüzü (UI) ile ilgili işlevleri içerir. ASP.NET Core MVC veya Razor Pages kullanılabilir. Kullanıcı girişleri alınır, iş katmanına istekler iletilir ve sonuçlar kullanıcı arayüzünde gösterilir. Bu katmanlar arasındaki bağlantılar, UI katmanının iş katmanına, iş katmanının veri erişim katmanına ve son olarak veri erişim katmanının varlık katmanına bağlandığı bir hiyerarşi oluşturur. Bu, uygulamanın bakımını kolaylaştırır ve her bir katmanın belirli bir sorumluluğu bulunmasını sağlar.



Şekil 2.1.4 UI Katmanı

2.2 Veri Tabanı

Veri tabanı, Entity Framework Core kullanılarak tasarlanmış ve yönetilmiştir. Migrations klasörü, veri tabanı şemasındaki değişiklikleri izlemek ve uygulamak için kullanılmıştır. Repository ve Context sınıfları, veri tabanı işlemlerinin merkezi yönetimini sağlamaktadır. Veri tabanı olarak Microsoft SQL Server (MSSQL) kullanılmıştır. Microsoft tarafından geliştirilen ve yaygın olarak kullanılan bir veritabanı yönetim sistemidir (RDBMS). MSSQL, tablo, sütun ve satırlar gibi standart bir ilişkisel veritabanı yapısını benimser. Tablolar, veritabanındaki veriyi düzenlemek için kullanılır ve sütunlar veritabanındaki özellikleri temsil eder. Her bir satır, bir kaydı veya bir ögeyi temsil eder.



Şekil 2.2 Veri Tabanı

2.3 Kullanılan Paketler

Projede kullanılan önemli paketler şunlardır:

Entity Framework Core: Entity Framework Core (EF Core), Microsoft tarafından geliştirilen ve .NET uygulamalarında veritabanı etkileşimini yönetmek için kullanılan bir nesne ilişkisel eşleme (ORM) framework'üdür. EF Core, .NET Core ve .NET 5 gibi çeşitli .NET platformlarına yöneliktir ve birçok veritabanı yönetim sistemini destekler. EF Core, nesne tabanlı programlamayı destekler ve veritabanı tablolarını .NET nesnelere ile eşleştirir. Bu sayede, veritabanı etkileşimi nesne yönelimli bir yaklaşıma dönüşür, ve geliştiriciler veritabanı işlemlerini nesne yöntemleriyle gerçekleştirebilir. EF Core, veritabanı şemalarını kod temelli migrasyonlar kullanarak yönetmeyi sağlar.

Entity Framework Core Design: Entity Framework Core Design paketi, Entity Framework Core (EF Core) kütüphanesini tasarım zamanındaki görevler için destekleyen bir bileşen içerir. Bu paket, EF Core ile ilgili tasarım zamanı görevleri için bir dizi komut ve araç sağlar.

Entity Framework Core Sql Server: Entity Framework Core SQL Server paketi, Entity Framework Core (EF Core) ile Microsoft SQL Server veritabanı için özel olarak geliştirilmiş bir sağlayıcıdır. Bu paket, EF Core'un genel yapılarını kullanarak, .NET uygulamalarının SQL Server veritabanlarıyla etkileşim kurmasını sağlar. SQL Server paketi, SQL Server veritabanına özgü özellikleri ve optimize edilmiş performansı destekler.

Entity Framework Core Tools: Entity Framework Core Tools, Entity Framework Core'un geliştirme ve yönetim görevlerini gerçekleştirmek için kullanılan bir dizi komut satırı aracını içeren bir pakettir. Bu araçlar, Entity Framework Core tabanlı projelerin geliştirme sürecini kolaylaştırır ve veritabanı işlemlerini yönetmeye yardımcı olur.

Fluent Validation: FluentValidation, .NET tabanlı projelerde giriş doğrulama (validation) kurallarını tanımlamak ve uygulamak için kullanılan bir kütüphanedir. Bu kütüphane, özellikle ASP.NET ve ASP.NET Core projelerinde kullanılmak üzere tasarlanmıştır ancak herhangi bir .NET uygulamasında kullanılabilir. Temel olarak, FluentValidation, nesne doğrulama kurallarını belirleme ve bu kuralları uygulama sürecini kolaylaştırmak için kullanılır. Giriş doğrulaması, kullanıcı tarafından sağlanan verilerin belirli kurallara uygun olup olmadığını kontrol etmek amacıyla gerçekleştirilir. FluentValidation, bu tür kuralları açıkça ve okunabilir bir şekilde tanımlama yeteneği sunar.

Visual Studio Web Code Generation Design: ASP.NET Core projelerinde web kodu oluşturma görevlerini kolaylaştırmak için kullanılan bir tasarım zamanı paketidir. Bu paket, ASP.NET Core MVC ve Razor Pages projelerinde kullanılan Scaffolding (İskeletleme) işlemlerini yönetir ve geliştiricilere önceden tanımlanmış şablonları kullanarak hızlıca web kodları oluşturma imkanı sunar.

2.4 Katmanlar Arası Referanslar

DataAccessLayer, sadece EntityLayer'ı referans gösterir.

BusinessLayer, EntityLayer ve DataAccessLayer'ı referans gösterir.

UI Katmanı, BusinessLayer, DataAccessLayer ve EntityLayer'ı referans gösterir.

2.5 Tema ve Tasarım

Projede iki adet hazır tema kullanılmıştır. "argon-dashboard-master" ve "corona-free-dark-bootstrap-admin-template-1.0.0" temaları, web sitesinin ve admin panelinin görsel tasarımını kolaylaştırmak için Türkçeleştirilmiş ve düzenlenmiştir. Temalar eklenirken kullanılan N-Tier Architecture mimarisine uygun olarak UI katmanına eklenmiştir. Okunabilirlik ve yönetilebilirlik adına MVC mimarisi bu katman içine dahil edilmiştir. Model-View-Controller (MVC), yazılım geliştirmenin bir tasarım desenidir ve genellikle kullanıcı arabirimi olan uygulamaların organizasyonunu kolaylaştırmak için kullanılır. Bu desen, uygulamayı üç temel bileşene ayırarak modüler ve sürdürülebilir bir yapı sunar. Model üzerinde işlem yapılmaya gerek duyulmadan Views ve Controller üzerinde daha çok işlem yapılmıştır.

View, kullanıcı arayüzünü temsil eder. Kullanıcıya gösterilen bilgilerin ve grafiksel öğelerin görsel tasarımını içerir. Kullanıcıya sunulan bilgiler genellikle Model tarafından sağlanır, ancak View, bu bilgileri düzenleyip gösterir. View, genellikle HTML, CSS ve diğer ön yüz teknolojilerini kullanarak kullanıcı arayüzünü oluşturur. Bir View, kullanıcı etkileşimine yanıt verebilir ve bu etkileşimlerin sonucunda Controller'a bildirimde bulunabilir.

Controller, kullanıcının etkileşimlerini yöneten ve Model ile View arasında iletişimi sağlayan bileşendir. Kullanıcının bir talebi olduğunda Controller bu talebi alır, ilgili iş mantığı veya veritabanı etkileşimlerini gerçekleştirir ve ardından View'e gerekli bilgileri iletir. Controller, kullanıcının etkileşimlerine göre Model'i güncelleyebilir veya View'ı değiştirebilir. Bu sayede uygulama içindeki mantıksal akışı kontrol eder. MVC, kullanıcı etkileşimlerini ve uygulama durumu değişikliklerini işlemek için Controller'ı kullanarak esnek bir yapı sunar. Temel olarak Model ise Model, uygulamanın veri ve iş mantığıyla ilgili kısımdır. Veritabanına erişim, veri manipülasyonu ve iş mantığı kurallarını içerir. Genellikle bu katman, uygulamanın içsel durumunu temsil eden sınıflar ve nesnelere içerir. Bu sınıflar, veritabanından veri almak, işlemek

ve saklamak gibi görevleri gerçekleştirebilir. Model, uygulamanın gerçek dünyadaki nesnelere temsil eder ve bu nesnelere ilgili işlemleri gerçekleştirir.

2.6 Partial View ve View Components

Partial View ve View Components, sayfaların okunmasını ve yönetilmesini kolaylaştırmak amacıyla kullanılmıştır. Kullanıcı ve admin paneli için bu yöntemler, karmaşık kodları daha anlaşılır ve yönetilebilir hale getirmektedir. Partial View, ASP.NET ve ASP.NET Core projelerinde kullanılan, bir sayfanın veya bir View'ın bir bölümünü temsil eden, kendi başına kullanılabilen ve başka bir View içine gömülebilen bir yapıdır. Genellikle, bir sayfanın belirli bir bölümünü veya bir View'ın tekrar kullanılabilir parçalarını temsil eden küçük parçalı görünümü oluşturmak için kullanılır. Belirli bir görünüm parçasını farklı sayfalarda veya bileşenlerde tekrar kullanabilirsiniz. Bir sayfa içindeki mantıksal parçaları ayrı görümlere bölebilir ve bu sayede kodunuzu daha düzenli ve bakımı daha kolay hale getirebilirsiniz. Bir sayfa içindeki sadece belirli bir bölümü güncellemek istediğinizde Partial View kullanabilir ve bu sayede sadece ilgili bölümü yeniden çizmeniz gerekir.

View Components, özellikle karmaşık ve tekrar kullanılabilir bileşenleri oluşturmak için kullanılan bir başka yapıdır. Partial View'dan farklı olarak, View Components, bir Controller sınıfından türetilir ve kendine ait iş mantığına sahiptir. View Components, sayfanın belirli bir bölümünü değil, genellikle bir widget veya daha karmaşık bir bileşeni temsil eder. View Components, kendi iş mantığı ve verilerini içerir ve bu sayede bağımsız çalışabilir. Bu, bir View Component'in bağlamından bağımsız olarak test edilebilir ve geliştirilebilir olmasını sağlar. View Components, kendi Model ve View'larına sahip olabilir, bu da onları daha özelleştirilebilir kılar. View Components, sadece basit HTML değil, aynı zamanda karmaşık iş mantığı içeren bileşenleri de temsil edebilir.

2.7 Fluent Validation

Fluent Validation, projede kullanıcı girişlerinin doğruluğunu kontrol etmek amacıyla entegre edilmiştir. Proje eklerken veya sertifika eklerken maddelerin boş geçilemez olmasını ve karakter sınırlamalarını sağlamak için kullanılmıştır.

FluentValidation, .NET tabanlı projelerde giriş doğrulama (validation) kurallarını tanımlamak ve uygulamak için kullanılan bir kütüphanedir. Bu kütüphane, özellikle ASP.NET ve ASP.NET Core projelerinde kullanılmak üzere tasarlanmıştır ancak herhangi bir .NET uygulamasında kullanılabilir. FluentValidation, kullanıcı girişi gibi verilerin doğruluğunu kontrol etmek için yaygın olarak kullanılır. Bu giriş doğrulama işlemleri, veri tabanına kaydetme öncesinde veya kullanıcıya hata mesajları gösterme amacıyla uygulanabilir.

2.8 Hata Yönetimi

Geliřtirdiđiniz web uygulamasında, kullanıcıların karşılařtıđı hataların yönetimi ve bu hatalara özelleřtirilmiř hata sayfalarının atanması, kullanıcı deneyimini olumlu yönde etkileyen kritik bir unsurdur. ASP.NET Core uygulamalarında, `app.UseStatusCodePagesWithReExecute` middleware'i ile HTTP durum kodlarına özel hata sayfalarını kolayca yönetebilmektedir. Bu middleware, belirli bir HTTP durum kodu alındığında belirli bir URL'ye yönlendirme yapar. Örneđin, 404 Not Found durum kodu alındığında kullanıcıyı özel bir 404 hata sayfasına yönlendirebilirsiniz. Bu, kullanıcılara daha anlamlı ve özelleřtirilmiř hata mesajları sunmanızı sađlar. Bu kullanım, kullanıcıların karşılařtıkları hatalara daha iyi anlamlar yükleyerek ve onları özel hata sayfalarına yönlendirerek uygulamanızın kullanıcı dostu olmasını sađlar.



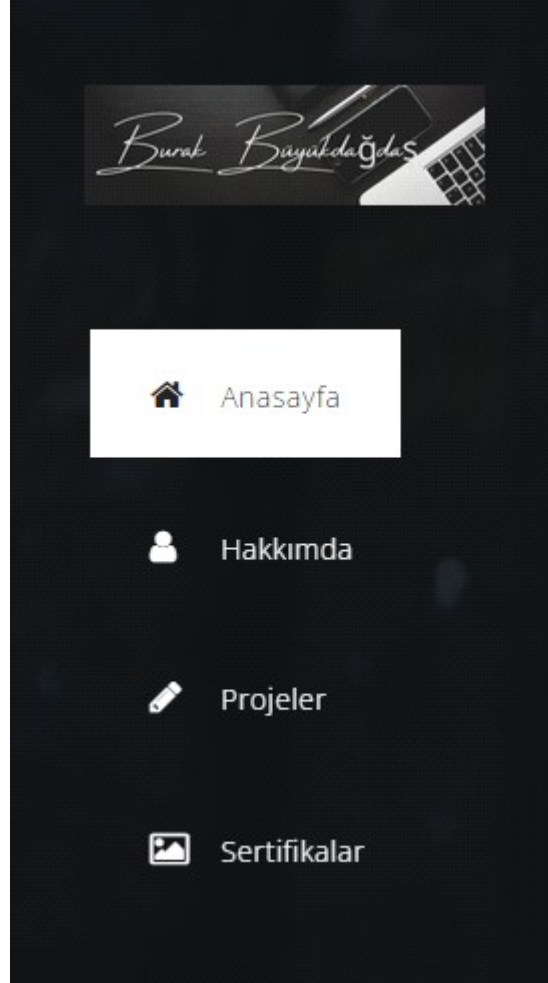
řekil 2.8 Hata Sayfası

Bölüm 3 Kullanıcı Yüzü

3.1 Menü:

Bu bölümde, web sitesinin sol panelinde konumlanan menü ve bu menünün altında yer alan görsel unsurların, kullanıcı deneyimine etkisini incelemeyi amaçlamaktadır. Menü, sitede kolay gezinme olanađı sađlamak adına temel bir bileřendir. Anasayfa, Hakkımda, Projeler ve Sertifikalar gibi ana bařlıklar, ziyaretçilere web sitesinin temel bölümlerine hızlı erişim imkanı tanır. Bu yapılandırma, içerikler arası geçişin kullanıcılar için sezgisel ve sorunsuz bir deneyim olmasını sađlar. Menünün hemen üst kısmında yer alan tasarlanmıř görsel, site temasına estetik bir katkı sađlamaktadır. Bu görsel unsurlar, ziyaretçilere sayfanın genel atmosferini yansıtarak,

görsel bir çekicilik oluşturur. Aynı zamanda, menüyle bütünleşen bir tasarım anlayışıyla, site içindeki içerikler arasında harmoniyi destekler.



Şekil 3.1 Menü Sayfası

3.2 Footer:

Web sitemin en altında konumlanan footer bölümü, ziyaretçilere çeşitli önemli bağlantılar ve iletişim bilgileri sunarak web sitesi deneyimini zenginleştiriyor. © 2023 Burak Büyükdağdaş yazısı, web sitesinin tüm içeriklerinin telif hakkına sahip olduğunu belirten bir bildiridir. Bu, ziyaretçilere içerikleri kullanırken belirli kullanım koşullarına uymalarını hatırlatır.

Whatsapp üzerinden kolaylıkla iletişime geçebilmek için Whatsapp İletişim bağlantısı, ziyaretçilere doğrudan bir iletişim kanalı sağlar. Bu bağlantı aracılığıyla, ziyaretçiler, herhangi bir soru sormak, görüş bildirmek veya işbirliği teklif etmek için hızlıca iletişime geçebilirler.

Projelerin daha detaylı incelenebilmesi için GitHub bağlantısı, ziyaretçilere projelerin kaynak kodlarına ve diğer detaylarına erişim imkanı sunar. Bu bağlantı, açık kaynak projeleriyle ilgilenen geliştiriciler ve diğer ziyaretçiler için faydalıdır.

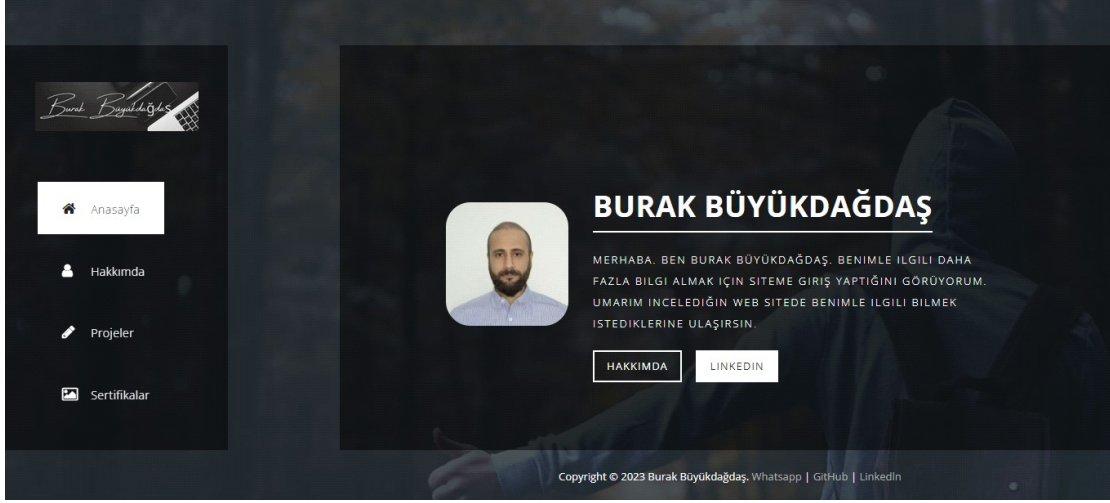
Sosyal ağ varlığını güçlendirmek ve iş bağlantılarını artırmak adına LinkedIn bağlantısı, ziyaretçilere profesyonel ağınızla bağlantı kurma imkanı sunar. Bu bağlantı aracılığıyla, iş dünyasındaki etkileşiminizi artırabilir ve yeni bağlantılar kurabilirsiniz.

Copyright © 2023 Burak Büyükdağdaş. Whatsapp | GitHub | LinkedIn

Şekil 3.2 Footer Sayfası

3.3 Anasayfa:

Geliştirdiğim kişisel portföy web sitesi, ziyaretçilere hoş bir karşılama sunuyor. Anasayfanın merkezinde, sade ve etkileyici bir tasarımla karşımıza çıkıyor. Web sitesinde Anasayfanın sol bölümünde Portföy sahibinin görseli yer almakta. Başlık bölümünde, ad ve soyadın vurgulanması, sayfanın ana karakterini belirlemekte ve ziyaretçilere, içeriğiyle ilgili temel bir bilgi vermektedir. Bu minimalist yaklaşım, sayfa tasarımının özünü basitleştirmekte ve odaklanmış bir kullanıcı deneyimi sağlamaktadır. Anasayfanın alt bölümünde 2 adet buton yer almakta. Bu butonlar sitenin 2. bölümü olan Hakkımda sayfasına yönlendirirken diğer buton ise Portföy sahibinin LinkedIn hesabına yönlendirmektedir.



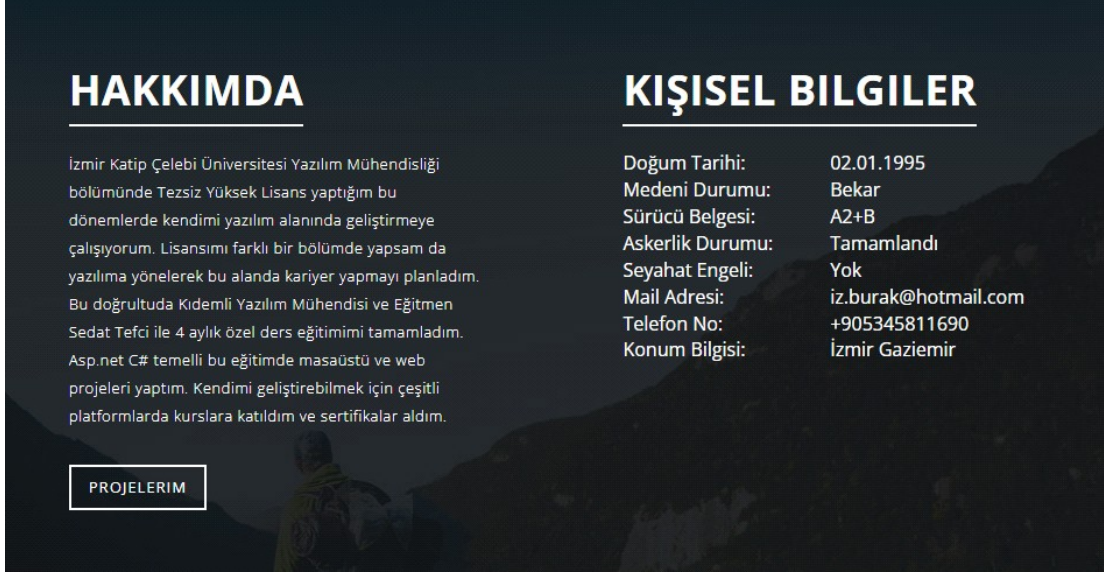
Şekil 3.3 Anasayfa

3.4 Hakkımda:

Hakkımda sayfası, web sitesinin merkezinde yer alan önemli bir bölümdür. Bu sayfa, portföy sahibinin kişisel ve profesyonel kimliğini ziyaretçilere tanıtmak için iki ayrı bölüme ayrılmıştır. Sol tarafta öne çıkan "HAKKIMDA" başlığı altında, portföy sahibi kendisini anlatan bir yazıya yer verilmiştir. Bu bölümde, kişisel özellikler, deneyimler ve tutkular hakkında detaylı bir içerik sunularak ziyaretçilere daha yakından tanıma fırsatı sağlanmıştır. Sağ tarafta ise "KİŞİSEL BİLGİLER" başlığı altında, portföy sahibinin daha özel detayları yer alır. Bu bölümde, doğum tarihi, medeni durumu, sürücü belgesi, askerlik durumu, seyahat engeli, mail adresi, telefon numarası, yaşadığı il ve ilçe bilgileri gibi kişisel ayrıntılar paylaşılmıştır. Bu bilgiler, ziyaretçilere portföy sahibinin kişisel yaşamına dair daha kapsamlı bir perspektif sunar.

Bu iki bölümlü anlatım, ziyaretçilere portföy sahibinin hem profesyonel hem de kişisel kimliğini eksiksiz bir şekilde tanıma şansı verir. Sayfanın bu yapılanması, hem genel profil hem de özel detaylarla bilgi sunarak ziyaretçilere kapsamlı ve etkileyici bir deneyim sunar.

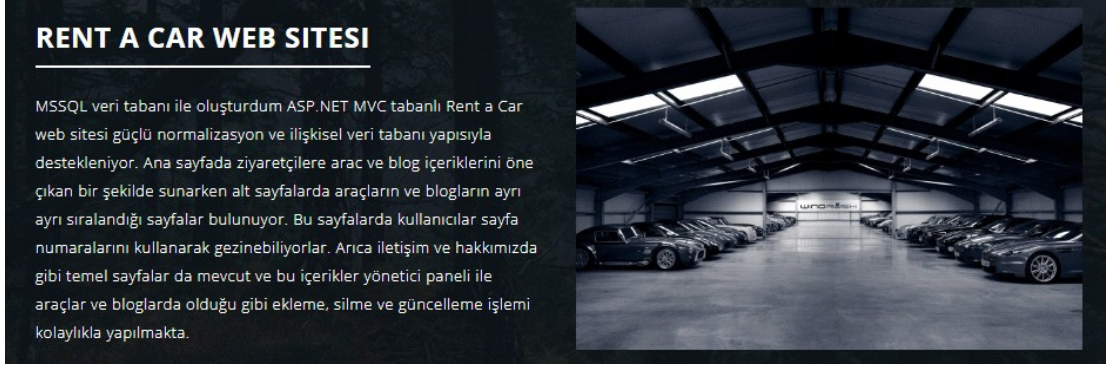
Bu bölümün hemen altında ise bir buton yer almaktadır. Bu buton Projelerim sayfasına yönlendirmek için kullanılır.



Şekil 3.4 Hakkımda Sayfası

3.5 Projeler:

Web sitesinin "Projeler" bölümü, portföy sahibinin yaratıcılığını, yeteneklerini ve çeşitliliğini ziyaretçilere sergileyen dinamik bir platform sunmaktadır. Bu bölüm, tamamen admin panelinden yönetilebilen bir altyapı ile projelerin eklenip çıkartılabildiği bir esnekliğe sahiptir. Bu bölümde yer alan projeler, geniş bir yelpazede çeşitlenmiştir. Her bir proje, başlığı, açıklaması ve projeye özgü bir görselle zenginleştirilmiştir. Yapılan her ekleme veya güncelleme, sitede anında yansıtılarak ziyaretçilere en güncel içeriği sunar. Admin paneli üzerinden projelerin eklenip çıkartılabilmesi, portföy sahibine büyük bir esneklik sağlar. Bu sayede, yeni projeler eklenebilir, var olanlar güncellenebilir veya belirli bir süre sonra gündemden kaldırılabilir. Bu dinamik yönetim, portföyün her zaman güncel ve çeşitli kalmasını sağlar. Her projenin yanında yer alan görseller, ziyaretçilere projelerin estetik ve işlevsel yönlerini gösterme imkanı sunar. Görseller, projelerin hemen yanında bulunduğu için ziyaretçiler, içeriği daha hızlı değerlendirir ve ilgi çeken projelere odaklanır. Projeler bölümü, web sitesinin dinamizmini ve portföy sahibinin yeteneklerini açıkça ortaya koyan bir vitrin olma özelliğine sahiptir. Bu bölüm, sadece içeriğin çeşitliliği ile değil, aynı zamanda yönetim kolaylığı ile de dikkat çeker ve ziyaretçilere interaktif bir deneyim sunar.



Şekil 3.5 Projelerim Sayfası

3.6 Sertifikalar:

Portföy web sitesinin son bölümü, hak kazanılmış sertifikaların öne çıkarıldığı ve görsel zenginliğin ön planda olduğu bir alan sunmaktadır. Bu bölüm, portföy sahibinin elde ettiği başarıları görsellerle sergileyerek, ziyaretçilere çeşitli yeteneklerini ve uzmanlığını daha yakından tanıma imkanı sunmaktadır. Sertifikalar sayfası, görsel ağırlıklı bir tasarıma sahiptir. Her bir sertifika, anlamlı bir görselle öne çıkarılmıştır. Bu görseller, ziyaretçilere hemen birinci bakışta sertifikaların niteliği hakkında bilgi verir ve ilgi çekici bir atmosfer oluşturur. Her bir sertifikaya ait görselin üzerine gidildiğinde, detaylar ortaya çıkar. Başlık ve açıklama, o sertifikaya dair bilgileri içerir. Bu detaylar, ziyaretçilere sertifikaların kazanılma sebepleri, içerikleri ve önemli detayları hakkında kapsamlı bir anlayış sunar. Sertifikalar sayfasında, ziyaretçiler sertifikalara daha yakından bakabilmek adına bir sertifikaya tıkladıklarında, görselin daha büyük bir versiyonunu görme imkanına sahiptir. Bu genişleme seçeneği, ziyaretçilere detaylı bir inceleme ve sertifikaların daha yakından değerlendirilmesi fırsatı sunar.

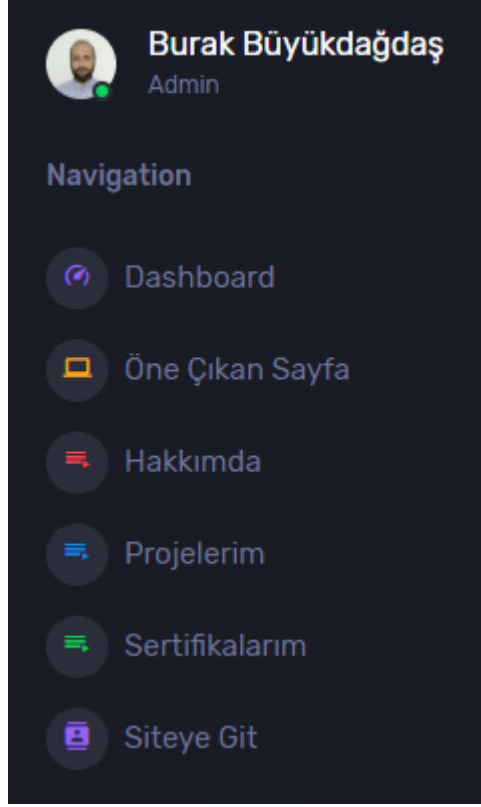


Şekil 3.6 Sertifikalarım Sayfası

Bölüm 4 Yönetici Paneli

4.1 Menü:

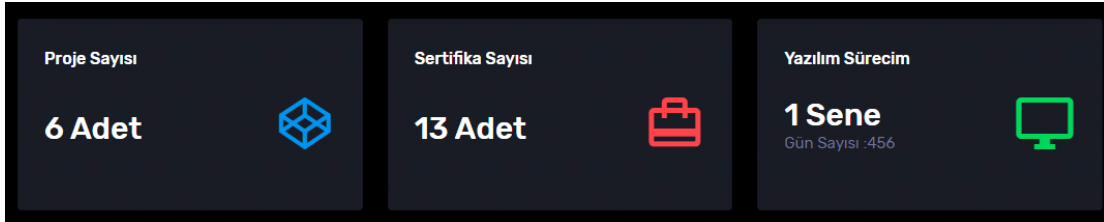
Portföy web sitesinin yönetilebilirliği için özel olarak oluşturulan admin paneli, kullanıcı dostu bir menü yapısına sahiptir. Bu menü, web sitesinin farklı bölümlerine hızlı ve kolay bir şekilde erişim sağlar. Bu menüde yer alan sayfalardan Dashboard, admin panelinin ana sayfasını temsil eder. Bu bölüm, genel istatistikleri, web sitesi performansını ve güncel durumu özetleyen bir görsel sunar. Portföy sahibinin web sitesiyle ilgili genel bakışını sağlamak adına önemli bir başlangıç noktasıdır. Öne Çıkan Sayfa, web sitesinin ana sayfasının düzenini belirleyen bir bölümdür. Hakkımda, portföy sahibinin kişisel bilgilerini ve profesyonel geçmişini düzenleyebileceği bir alanı ifade eder. Projelerim, web sitesinde sergilenen projelerin yönetimini sağlar. Sertifikalarım, web sitesinde sergilenen sertifikaların yönetimini sağlar. Siteye Git seçeneği, web sitesinin canlı haline hızlı bir şekilde yönlendirme yapar.



Şekil 4.1 Yönetici Menü Sayfası

4.2 Dashboard:

En üst bölümde yer alan 3 adet tabloda, proje sayısı, sertifika sayısı ve portföy sahibinin ilgilendiği iş için geçen süre detaylı bir şekilde sunulmaktadır. Bu istatistikler, web sitesinin genel performansına dair anlık bir bakış sağlar.



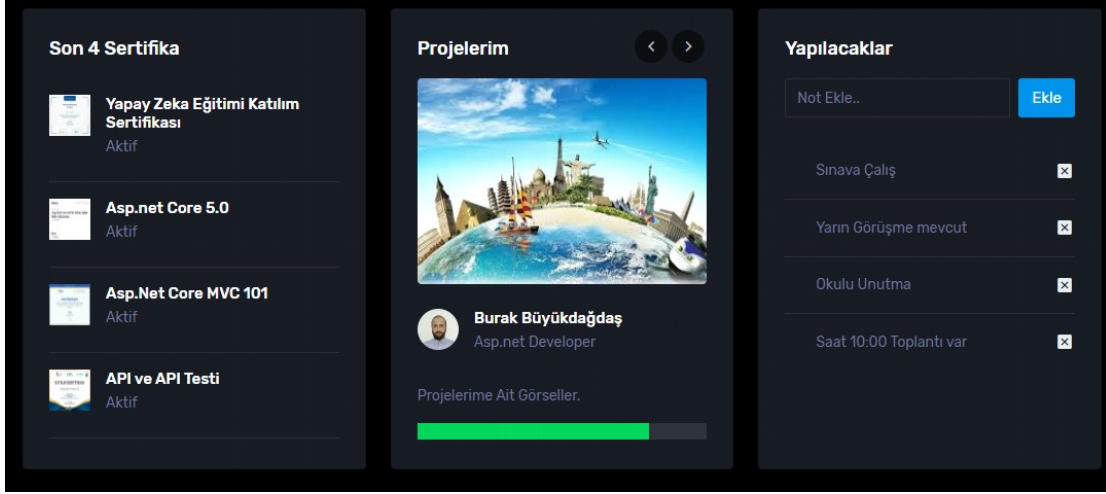
Şekil 4.2.1 Dashboard Sayfası

Hemen altında, sertifikalar ile ilgili istatistikleri içeren bir bölüm yer alır. Bu tablo, toplam sertifika sayısı, tamamlananlar ve devam eden sertifikalar gibi detayları içerir. Bu istatistikler, portföy sahibinin profesyonel gelişimini izlemek adına önemlidir. Yanında ise son 4 projeye ve ait görseller, başlıklar ve açıklamalar bulunmaktadır. Bu bölüm, portföy sahibinin en güncel projeleri ve sertifikaları hızlıca keşfetme imkanı tanır.



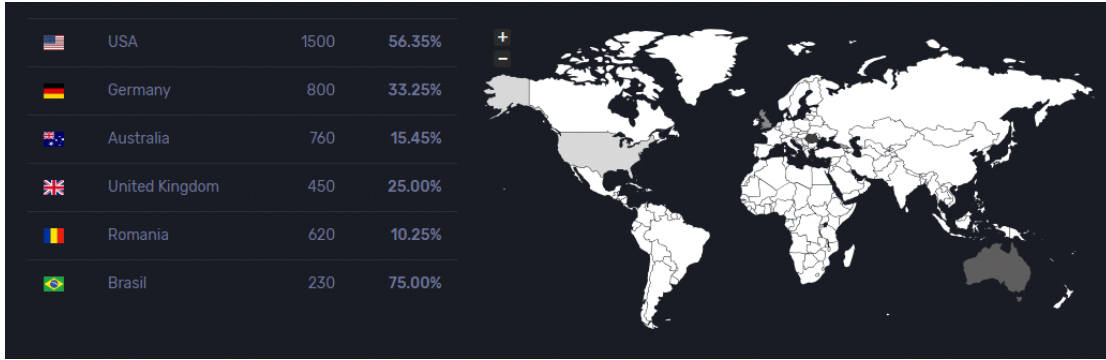
Şekil 4.2.2 Dashboard Sertifika ve Proje Listeleme Sayfası

Bu bölümün altında ise son 4 sertifikaya ait görseller, başlıklar, açıklamalar ve Sertifika durumunu gösteren yazı bulunmaktadır. Yanında portföy sahibinin çalışma alanını belirten bir görsel ve hemen yanında yer alan "Yapılacaklar" bölümü bulunur. Bu bölüm, portföy sahibinin notlarını tutabileceği, projeleri ve işleri yönetebileceği bir alandır.



Şekil 4.2.3 Dashboard Sertifika,Proje ve Yapılacaklar Listeleme Sayfası

Son olarakta en alt kısımda, dünya haritası yer almaktadır. Bu bölüm daha çok görsel amaçlı bulunmaktadır.

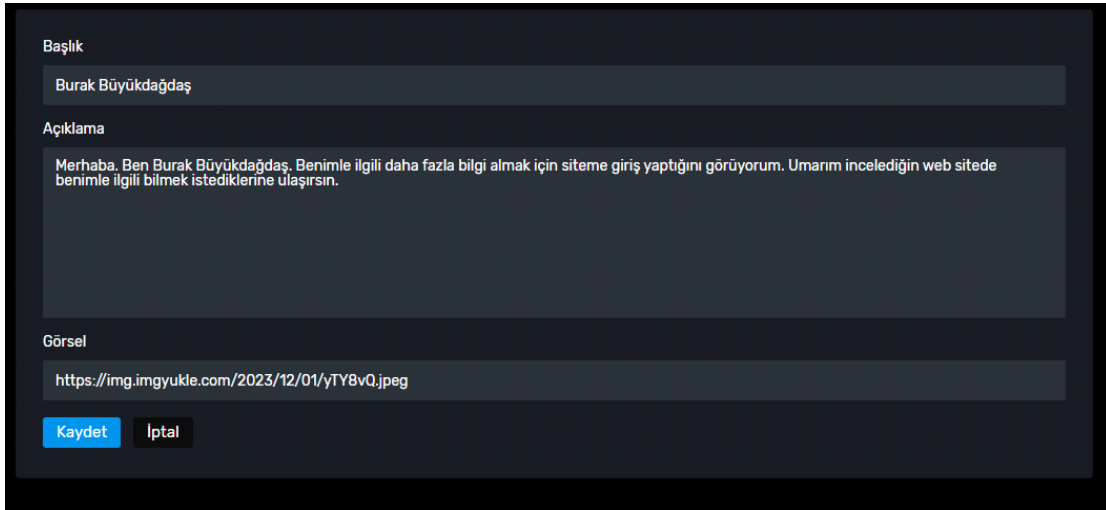


Şekil 4.2.4 Dashboard Temsili Harita Sayfası

4.3 Öne Çıkan Sayfa:

Admin panelindeki "Öne Çıkan Sayfa" bölümü, kullanıcı sayfanızın ana sayfasındaki başlık, açıklama ve görseli düzenlemenizi sağlayan kullanıcı dostu bir arayüze sahiptir. Bu bölüm, web sitenizin ilk izlenimi olan ana sayfa içeriğini özelleştirmenize olanak tanır. Bu bölümde, ana

sayfanızdaki başlık, açıklama ve görsel içeriği anlık olarak değiştirebilirsiniz. Her bir öge için ayrı bir düzenleme alanı bulunur, böylece istediğiniz öğeyi hızlıca güncelleyebilirsiniz. Değişikliklerinizi tamamladıktan sonra, "Kaydet" butonuyla yeni içeriği uygulayabilirsiniz. Eğer değişiklikleri iptal etmek isterseniz, "İptal" butonunu kullanabilir ve mevcut içeriği koruyabilirsiniz. Bu basit ve kullanıcı dostu düzenleme arayüzü sayesinde, ana sayfanızın içeriğini güncellemek ve web sitenizi dinamik tutmak artık çok daha kolay. "Öne Çıkan Sayfa" bölümü, içerik yönetimini hızlı, etkili ve kolay bir hale getirir, böylece ziyaretçilerinize her zaman güncel ve ilgi çekici bir deneyim sunabilirsiniz.



Şekil 4.3 Öne Çıkan Sayfas

4.4 Hakkımda:

Admin panelinizdeki "Hakkımda" bölümü, kullanıcı sayfanızda yer alan "Hakkımda" kısmını kişiselleştirmenizi sağlayan pratik bir araçtır. Bu bölüm, ziyaretçilerinize kendinizi en iyi şekilde tanıtabilmeniz için tasarlanmıştır. Bu bölümde, "Hakkımda" sayfanızdaki başlık ve açıklama alanlarını kolayca değiştirebilirsiniz. Her iki alanda da metin düzenleme seçenekleri bulunur, bu da içeriğinizi istediğiniz gibi özelleştirmenize olanak tanır. Yaptığınız değişiklikleri uygulamak için "Kaydet" butonunu kullanabilirsiniz. Eğer değişiklikleri iptal etmek isterseniz, "İptal" butonu mevcut içeriği korumanıza yardımcı olur. Bu basit ve etkili düzenleme arayüzü sayesinde, "Hakkımda" bölümünüzü güncellemek ve ziyaretçilerinize kendinizi en iyi şekilde tanıtmak artık çok daha kolay. Admin panelinizdeki bu özellik, kullanıcı sayfanızı güncel tutmanıza ve kişisel markanızı daha çekici hale getirmenize yardımcı olur.

Başlık

Hakkımda

Açıklaması







Izmir Katip Çelebi Üniversitesi Yazılım Mühendisliği bölümünde Tezsiz Yüksek Lisans yaptığım bu dönemlerde kendimi yazılım alanında geliştirmeye çalışıyorum. Lisansımı farklı bir bölümde yapsam da yazılıma yönelerek bu alanda kariyer yapmayı planladım. Bu doğrultuda Kıdemli Yazılım Mühendisi ve Eğitimci Sedat Tefci ile 4 aylık özel ders eğitimimi tamamladım. Asp.net C# temelli bu eğitimde masaüstü ve web projeleri yaptım. Kendimi geliştirebilmek için çeşitli platformlarda kurslara katıldım ve sertifikalar aldım.

Kaydet İptal

Şekil 4.4 Yönetici Hakkımda Sayfası

4.5 Projelerim:


Bu bölümde, mevcut projelerinizin başlıkları, açıklamaları, görselleri, silme ve düzenleme butonlarıyla birlikte liste halinde görüntülenir. Her bir proje, "Sil" ve "Düzenle" butonları aracılığıyla kolayca yönetilebilir. Her bir proje, altında başlık, açıklama ve görsel ile detaylı bir şekilde listelenir. Bu detaylar, projelerinizi ziyaretçilere daha ayrıntılı bir şekilde tanıtmak için kullanılır.

Başlık	Açıklama	Resim	Sil	Düzenle
Derin Öğrenme	Pyhton'da derin öğre ...		Sil	Düzenle
Rent a Car Web Sitesi	MSSQL veri tabanı il ...		Sil	Düzenle
Veri Analizi	Pyhton kullanarak Pe ...		Sil	Düzenle
E-Ticaret web Sitesi	Asp.Net MVC ile admi ...		Sil	Düzenle
Portfolio Web Site	Kişisel olarak dinam ...		Sil	Düzenle
Seyahat ve Yemek Web Sitesi	Asp.net mvc ile oluş ...		Sil	Düzenle

Proje Ekle

Şekil 4.5.1 Yönetici Projelerim Sayfası

Bu sayfanın hemen altında bulunan "Proje Ekle" butonu, yeni projeler eklemek için hızlı bir erişim sağlar. Bu butona tıkladığınızda, yeni bir proje eklemek için gerekli bilgileri kolayca girebilir ve web sitenizi güncel tutulabilmeye olanak sağlar.



The image shows a dark-themed form for adding a project. It contains three main input areas: 'Proje Başlığı' (Project Title), 'Sertifika Proje Açıklaması' (Certificate Project Description), and 'Proje Görseli' (Project Image). At the bottom left, there are two buttons: 'Kaydet' (Save) in blue and 'İptal' (Cancel) in grey.

Şekil 4.5.2 Yönetici Projelerim Ekleme Sayfası

4.6 Sertifikalarım:

Admin panelinizdeki "Sertifikalarım" bölümü, kişisel niteliklerinizi ve başarılarınızı düzenlemenizi sağlayan kullanıcı dostu bir arayüze sahiptir. Bu bölüm, sertifikalarınızı listeler, güncelleme ve yeni sertifikalar eklemenize olanak tanır. Bu bölümde, mevcut sertifikalarınızın adları, açıklamaları, durumları, görselleri, tamamlanma durumları ve silme/düzenleme butonlarıyla birlikte listelenir. Her bir sertifika, detaylı bilgileriyle birlikte kolayca yönetilebilir. Her bir sertifika, altında adı, açıklaması, durumu, görseli ve tamamlanma durumu ile detaylı bir şekilde listelenir. Bu detaylar, ziyaretçilere sertifikalarınızın önemini ve değerini daha iyi anlatmanıza yardımcı olur.

Sertifika	Açıklama	Durum	Sertifika	Tamamlama	Sil	
	C# Katılım Belgesi	Engin Demiroğ ...	Aktif		Tamamlandı	Sil
	Web Sitesi Kullanılabilirliği	Kullanıcı dost ...	Aktif		Tamamlandı	Sil
	C# ile Algoritma ve Programlama 101	Murat Yücedağ ...	Aktif		Tamamlandı	Sil
	C# ile Algoritma ve Programlama 201	Murat Yücedağ ...	Aktif		Tamamlandı	Sil
	"C# Form ile Görsel ve Nesne Tabanlı Programlama 101	Murat Yücedağ ...	Aktif		Tamamlandı	Sil
	C# FORM İLE GÖRSEL VE NESNE TABANLI PROGRAMLAMA 201	MURAT YÜCEDAG ...	Aktif		Tamamlandı	Sil
	C# FORM İLE GÖRSEL VE NESNE TABANLI PROGRAMLAMA 301	MURAT YÜCEDAG ...	Aktif		Tamamlandı	Sil

Şekil 4.6.1 Yönetici Sertifikalarım Sayfası

Listemele işleminin hemen altında yer alan "Sertifika Ekle" butonu, yeni sertifikalar eklemek için hızlı bir erişim sağlar. Bu butona tıkladığınızda, yeni bir sertifika eklemek için gerekli bilgileri kolayca girebilir ve kişisel başarılarınızı güncelleyebilirsiniz.

Sertifika Platformu

Sertifika Başlığı

Sertifika Açıklaması

Sertifika Durumu

Sertifika Görseli

Sertifika Tamamlanması

[Kaydet](#) [İptal](#)

Şekil 4.6.2 Yönetici Sertifika Ekleme Sayfası

Bu kullanıcı dostu "Sertifikalarım" bölümü sayesinde, kişisel niteliklerinizi ve başarılarınızı düzenlemek ve ziyaretçilere göstermek artık daha etkili ve basit. Admin panelinizin bu özelliği, sertifikalarınızı ve niteliklerinizi güncel tutmanıza yardımcı olur.

4.7 Siteye Git:

Siteye Git seçeneği, web sitesinin canlı haline hızlı bir şekilde yönlendirme yapar. Bu seçenek, admin panelinde yapılan değişiklikleri canlı web sitesinde görmek isteyen kullanıcılara pratik bir erişim sunar.

4.8 Footer:

Admin panelinizin "Footer" bölümü, web sitenizin alt kısmında yer alan bilgileri özelleştirmenizi sağlar. Bu bölüm, ziyaretçilere tarih bilgisi ve iletişim detayları sunarak web sitenizin güvenilirliğini artırır ve tarih bilgisi verir.

Copyright © Burak Büyükdağdaş 2023

Şekil 4.8 Yönetici Footer Sayfası

Bölüm 5 Web Sitenin İnternete Atılması

5.1 Domain-Hosting-Veri Tabanı işlemler:

Domain, internet üzerindeki bir kaynağı (genellikle bir web sitesini) tanımlayan, hatırlanması kolay bir adrestir. İnternette dolaşırken, web sitelerine erişmek için IP adresleri kullanılır, ancak IP adresleri karmaşık ve hatırlanması zor olduğundan, domain adları bu işlevi yerine getirir.

Örneğin, "https://ikcu.edu.tr" bir domain adıdır. Domain adları, genellikle "www" (World Wide Web) ile başlar, ardından sitenin adı gelir ve en sonunda "com", "net", "org" gibi üst düzey etki alanı uzantıları bulunur. Bu uzantılar, genellikle sitenin türünü veya coğrafi konumunu belirtir.



Şekil 5.1 Domain ve Hosting

Hosting, bir web sitesinin veya uygulamanın internet üzerinde erişilebilir olabilmesi için gerekli olan altyapıyı sağlayan hizmettir. Bir web sitesi oluşturduğunuzda, bu sitenin dosyaları (HTML, CSS, resimler vb.) bir sunucuda depolanır. Bu sunucu, web sitenizin internet üzerinde görüntülenmesini mümkün kılar.

Bir web sitesi oluşturduğunuzda, bir domain adı ve hosting hizmetine ihtiyaç duyulur. Domain adı, sitenize insanların erişebilmesi için kullanılan adrestir; hosting ise web sitenizin dosyalarını depoladığımız altyapıdır. İyi bir hosting sağlayıcısı, sitenizin hızlı, güvenli ve sürekli olarak erişilebilir olmasını sağlamak için önemlidir.

Domain ve hosting hizmetleri birçok şirket tarafından sağlanmaktadır. Bu hizmetleri alırken güvenilir, hızlı, güvenli ve müşteri desteği sağlayan bir sağlayıcı seçmek önemlidir. Bu şirketler ücretli ve ücretsiz olarak hizmet verebilmektedir.

Web sitesi veya uygulamanız için uygun domain ve hosting hizmeti seçerken, kullanmayı planladığınız veritabanı türü büyük bir öneme sahiptir. Özellikle MSSQL veritabanı kullananlar için, doğru domain ve hosting sağlayıcısını seçmek başarı için kritik bir adımdır.

MSSQL veritabanınızı kullanırken, domain adınızın bu teknolojiyle uyumlu olduğundan emin olmalısınız. Domain kaydınızı yaparken, sağlayıcınızın MSSQL ile entegrasyonunu destekleyip desteklemediğini kontrol etmek önemlidir. Önde gelen domain sağlayıcıları genellikle bu konuda destek sunar ve domaininizin DNS ayarlarını MSSQL veritabanınıza yönlendirmenize izin verir. Satın alınan şirkette veri tabanı sunucularına bağlanıp projeyi buna uyumlu yapılması gerekmektedir.

5.2 Publish İşlemi:

Asp.net Core projenizi internette yayınlamak, projenizi geniş bir kullanıcı kitlesiyle paylaşmanın ve erişilebilir kılmanın kritik bir adımdır. Publish işleminden önce proje detaylı olarak test edilmeli ve test sonrası temizlenip derleme işlemi yapılmalıdır. Bu işlem sonrasında Solution Explorer üzerinden projenize sağ tıklayın ve "Publish" seçeneğini seçilmelidir. Folder ayarları sonrasında işleme başlanabilir.



Şekil 5.2 Publish

5.3 FileZilla ile Paylaşım:

Publish işlemi ardından web sitesini canlıya taşımak için FileZilla gibi uygulamalar kullanılmadığıdır. FileZilla indirilip açıldıktan sonra Domain ve Hosting tedarik ettiğimiz şirkete girip Sunucu, Kullanıcı Adı ve parola bilgileri alarak giriş yapılmalıdır.

Sunucu:	<input type="text"/>	Kullanıcı adı:	<input type="text"/>	Parola:	<input type="text"/>	Bağlantı noktası:	<input type="text"/>	<input type="button" value="Hızlı bağlan"/>
---------	----------------------	----------------	----------------------	---------	----------------------	-------------------	----------------------	---

Şekil 5.3 FileZilla Panel Girişi

Giriş başarı ile yapıldıktan sonra publish edilen dosya seçilip dosyanın tamamı canlıya taşınması için sunucuya yüklenmelidir. Projeye ait tüm dosyaların yüklenmesi biraz zaman alsada parça parça yüklenme işlemi tamamlanıp canlıya taşınmış olacaktır.

Kaynaklar

Ateş, V., Karacan, H. (2010). Abant İzzet Baysal Üniversitesi web sitesi kullanılabilirlik analizi. *International Journal Of Informatics Technologies*, 2(2).

Ateş, U. ve Boz, M. (2015). Konaklama işletmelerinin web sitelerinin değerlendirilmesi: Çanakkale örneği. *Ekonomi, İşletme, Siyaset ve Uluslararası İlişkiler Dergisi*, 1 (1), 63-84.

ÇATAL D, KÜRŞAD D (June 1, 2016) Duyarlı Web Tasarımı. *Art-e Sanat Dergisi* 8 15 95–109.

Aral, S. (2011). Seo ile webde maksimum erişirlik. İstanbul: Pusula Yayınları.

Akçay M., Kasım Ö., Taşdelen Z. ASP.NET Ve MVC Temelli Esnek (Responsive) Web Uygulaması, *ESTUDAM Bilişim Dergisi*. - 2021. - Cilt 2, Sayı 1. – S. 34-41.

Ergin İ., Akseki B. Lisanüstü Eğitimde Kullanılan Öğrenci Bilgi Sistemi, *Eğitim ve Öğretim Araştırmaları Dergisi*. – 2012. – Cilt 1 Sayı 2. – S. 364-380.

Akçay M., Kasım O., Taşdelen Z. ASP.NET Ve MVC Temelli Esnek (Responsive) Web Uygulaması, *Journal of ESTUDAM Information*. - 2021. - Volume 2, Issue 1. – P. 34-41.

G. Cantürk , "Etkileşimli Tahtalarda Açık Kaynak Kod Tabanlı Pardon İşletim Sisteminin Kullanımına Yönelik Öğretmen Görüşleri", *Uluslararası Karamanoğlu Mehmetbey Eğitim Araştırmaları Dergisi*, c. 2, sayı. 1, ss. 48-61, Haz. 2020

Armutlu, H., Armutlu, Ş., & Akçay, M., (2012). İyi Bir Web Sitesi Nasıl Yapılır? XIV. Akademik Bilişim Konferansı.

Gürbüz A. Yazılım Test Mühendisliği. 1. Baskı. Ankara, Türkiye, Papatya Yayıncılık Eğitim, 2010.

Erdem O.A, Younis A.E. “Yazılım Projelerinde Risk Yönetimi”. *International Journal of Informatics Technologies*, 5(1), 1-6, 2012.

Ekren, G. & Turan, A. H. (2021). Yazılım geliştirme yaşam döngüsü bakış açısından ERP uygulaması: Bir delphi çalışması. İşletme Araştırmaları Dergisi, 13(1), 667-686.